

The Debit=Credit group structure, How the balance constraint brings structure to the books.

Sander Renes[†]

Delft University of Technology

January 7, 2026

Abstract

This paper constructs a mathematical description of the bookkeeping system and derives the properties of this system. The balance constraint (debit=credit) is shown to have equivalent effects when posed on journal entries or on the entire system. I show this constraint allows us to check and correct books, derive the accounting equations, and impute information. These fundamental procedures thought in introductory accounting courses are shown to stem from one common source, which should help their explanation. The common source of these bookkeeping processes highlights that how we record information in bookkeeping impacts the entire accounting process. It is also shown that debits and credits, matching and even accruals are not necessary for the relation between the balance constraint and the three procedures to exist, meaning that double-entry accounting consists of several components that could be separated in the design of the accounting and bookkeeping system.

This paper is derived from early versions of other papers on the mathematics of bookkeeping. It first repeats some of the required notation and main results, it's main value is to show the link between the math of bookkeeping and some of the material covered in basic bookkeeping and/or accounting courses. This paper has benefited greatly from discussions with Jilde Garst, Jeroen Suijs, David Ellerman and seminar audiences at the Erasmus University Rotterdam, all remaining errors are my own.

[†]s.renes-1@tudelft.nl

1 Introduction

Accounting for an economic event occurs in three steps, first we translate the event (or transaction) into accounting definitions, second we define the value effects of this event, third we record the event. The first two steps deal with the definitions of what needs to be registered and against what value, this is covered by the accounting definitions within IFRS or local GAAP and usually grouped under valuation. The third step, the actual bookkeeping is the topic of this paper.

The first recorded explanation of a double-entry bookkeeping system is found in book 9 of Pacioli (1494).¹ Pacioli's book is remarkable not just because it contains the first written explanation of the bookkeeping process and develops a new way to teach bookkeeping from basic principles, but also because it takes the time to summarize a great deal of mathematics (Sangster 2017). Pacioli goes as far to say that bookkeeping cannot be done without this knowledge, showing what was apparently accepted as an important pre-condition for accounting in the 15th century: know your math! This paper revisits this link between mathematics and accounting once more, by studying how the mathematical structure of bookkeeping defines certain accounting procedures.

Bookkeeping is an important aspect of economics, and in particular for accounting, for several reasons. First, bookkeeping is the primary technology for accounting and a primary source of data. It is where a significant part (if not most) of accounting data is recorded. This data is later used for decision making in companies, to decide on stock investments by investors, to supply to the tax authorities with information, to derive the national income accounts, etc. The bookkeeping system of companies is a primary source of data in all of economics. The better we understand this process, the better we can understand the data that it generates. Second, bookkeeping is the last step in the accounting process of economic events. This means that if we want to optimize, or simply improve the design of the accounting process and the data it generates, Bellman's principle tells us we should start with optimizing the bookkeeping process.² To see the impact bookkeeping has on accounting and accounting education, one only has to try to imagine what would happen without bookkeeping. If you do not have or do not understand the data, what can you do? As a consequence, bookkeeping is the first thing we teach students in accounting. Everything else has to wait until the recording process is clear.

Bookkeeping has a strict set of rules for changing the accounts. We only add those items together that belong together, i.e. we make sure we note the correct name of each account and do not accidentally add the new Accounts payable to the Accounts receivable. Furthermore, we keep good track of what value is in debit and what value is in credit, so that we increase or decrease the accounts as required. As long as these rules are followed, we can transform one set of valid financial statements into another (updated) set of valid financial statements. An operation that takes two elements of a set (two financial statements) and transforms them into a third element from the same set (updated financial statements) is analogous to the mathematical structure of a group law operating on its associated group set. Journal entries, general ledgers and balance sheets are all elements of a set of valid financial statements and we can define

¹A modern translation of the bookkeeping part of the treatise can be found in Cripps (1995)

²The same principle is the reason we start the budgeting process at the sales budget. In any type of planning, it is optimal to work backwards from our final goals.

processing of journal entries as a consistent group law. In this paper I build on the group-theoretical work of Ellerman (1985, 2014) and connect double-entry bookkeeping to the underlying mathematical structure of an Abelian group that concisely describes the current system.

This paper shows that the requirement that debit=credit (further: balance constraint) critically determines the usefulness of this group and our bookkeeping system. The main result shows that the requirement that debit=credit over the entire set of accounts, implies that every journal entry made has to be balanced. The reverse result is also shown to hold. As long as we start from an empty ledger, if we require every journal entry to be balanced, the resulting ledger will be balanced. Thus you cannot have one without the other. The result has several corollaries that show how we use this constraint in several procedures in bookkeeping, namely when we check our work with a trial balance, how we define the accounting equations to see what is debit and what is credit, and to impute missing information. This brings these common procedures in accounting together in a common frame and allows us to relate these procedures to each other.

In a common frame it is easier to transfers solution strategies and lessons learned in one setting to others. This has been shown to increase understanding and problem solving abilities in mathematical word problems, see Gvozdic and Sander (2019) and references therein. Word problems involve translating stories with numbers into a mathematical structure and solving the resulting mathematical problem. A similar process is involved when translating economic transactions to bookkeeping transactions, such that an abstract frame to relate the bookkeeping problems through might offer similar benefits.

These results also allows us to look at how we define double-entry bookkeeping. A lot of the properties of modern bookkeeping relate to the balance constraint. However, the use of debits and credits and of temporal matching, two aspects of bookkeeping that are often conflated with the balance constraint under the term ‘double-entry bookkeeping’, are in fact not required or implied by this constraint. In theory we could even design bookkeeping without these properties, and still keep many of the properties of modern bookkeeping.

In the following sections I show the properties of the group used in double-entry bookkeeping and the crucial role played by the balance constraint (Debits=Credits) in bookkeeping.

2 A mathematical description of bookkeeping

To build the accounting system, we first define the smallest element. This is just an element in which we can record some value, defined formally. Since our bookkeeping system keeps track of many different values, we will build a larger system with more than one of these elements later.

First, define an element consisting of a tuple of two numbers, $\alpha = [d // c]$, where d is a debit value and c is a credit value, both expressed as a non-negative number. The box is used to clearly separate elements, while the double slashes are used to separate debit and credit entries in an element. Denote the set off all such elements as \mathcal{A} :

$$\mathcal{A} = \{[d // c] | d, c \geq 0, d, c \in \mathcal{R}\}$$

If we take two of these α -elements, $\alpha_1 = [d_1//c_1]$, $\alpha_2 = [d_2//c_2]$, we can add them together by separately adding debit and credit values: $\alpha_1 +_{dc} \alpha_2 = [d_1 + d_2 // c_1 + c_2] = \alpha_3$. Since α_3 is a tuple with 2 non-negative numbers, $\alpha_3 \in \mathcal{A}$. Here we use $+_{dc}$ to denote the operation of separately adding the debit and credit values, the subscript indicates that we have to keep the debit and credit separated. Any account in the bookkeeping system can be associated with such an α -element. For an inventory, the debit entry would register the sum of all additions to the inventory, the credit entry the sum of all outflows out of the inventory. The current value in the inventory is found by taking the difference between debit and credit values registered in the account.

This definition of value – the difference between debit and credit value in a given account – implies that two α -elements have the same value if the difference between their debit and credit entries is the same. So in our inventory example, for the value of the inventory it is equivalent if the inventory value of 10 is obtained by debiting it for 60 and crediting it for 50, or it is obtained by debiting it for 100 and crediting it by 90. That is, we say the value of two of these elements is the same, and thus two of these elements are equivalent (denoted by \simeq) if the difference between their debit and credit values is the same: $[60//50] \simeq [100//90]$.

A binary operation that takes two elements from a set (the α -elements here) and yields a third element from the same set is what defines a group law. A set of elements combined with a group law defines a group, so that these α -elements together with the group law of addition of debit and credit values forms a group that I will refer to as the \mathcal{A} -group. Because the equivalence relation defined over the α -elements means that each element is valued as the difference between the two entries, this group is known as the group of differences in mathematics. This group has also been dubbed the Pacioli-group (Ellerman 1985, 2014) due to it's link with bookkeeping.

Lemma 1. *the \mathcal{A} -group (or Pacioli-group), $\{\mathcal{A}, +_{dc}\}$, consisting of elements $\alpha \in \mathcal{A}$ and the group law of addition of debit and credit values, $+_{dc}$, forms a group.*

The appendix shows that the following four necessary and sufficient properties are met: closure; associativity; existence of an identity element in the set; and existence of inverse elements for every element in the set. We will briefly explain the requirements and their meaning in the example of an inventory account that has an associated α -element to keep track of its value. Closure is defined mathematically as $\alpha_1 +_{dc} \alpha_2 \simeq \alpha_3 \in \mathcal{A} \quad \forall \alpha_1, \alpha_2 \in \mathcal{A}$. For an inventory it means that if we (correctly) add the debit and the credit value (the debit value in the second α -element) to our inventory (the value in the first α -element), the resulting inventory value is possible in our system. I.e. if we stay within the rules of bookkeeping, we always end up with a valid bookkeeping statement. Associativity requires $(\alpha_1 +_{dc} \alpha_2) +_{dc} \alpha_3 \simeq \alpha_1 +_{dc} (\alpha_2 +_{dc} \alpha_3)$, such that the order of operations does not matter. This holds in our inventory example since the order of processing transactions does not change the final result of all transactions. If we first debit the inventory for 60 and then credit it for 50, the total debit value increases by 10. Similarly, if we first credit it for 50 and then debit it for 60 this also increases the debit value by 10. Hence bookkeeping satisfies associativity. The existence of an identity element is formally defined as: $\exists e \text{ s.t. } \alpha + e \simeq \alpha \quad \forall \alpha \in \mathcal{A}$. There has to be some element in the group set that does not change the value of the other elements. Clearly, this element has to be a zero element: $e = [0//0]$. If we debit or credit our inventory by 0, this does not change

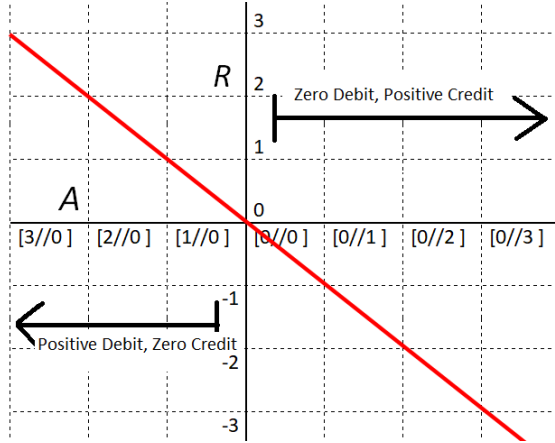


Figure 1. Bijection between set of \mathcal{A} -accounts and real line.

the inventory value. This identity element is part of our set, $[0//0] \in \mathcal{A}$, since it consists of a tuple of two non-negative numbers as required. Similarly, the presence of an inverse element is easily shown. Formally, an inverse element (denoted by superscript -1) is defined as $\forall \alpha \in \mathcal{A} \exists \alpha^{-1} \text{ s.t. } \alpha +_{dc} \alpha^{-1} = e$. In our setting the element e is an element without debit or credit value, like the empty inventory. To set an inventory to zero value, we simply credit it for the existing debit value. We can do the same to any account by simply applying the group law to the account and a second account where we exchange the debit and credit values. By exchanging debit and credit values, we create an inverse element that is a tuple with two non-negative numbers, so the inverse of $\alpha_1 = [d_1//c_1]$ is simply $\alpha_1^{-1} = [c_1//d_1] \in \mathcal{A}$. These 4 properties of the \mathcal{A} -group also provide a simple algebra that one can use to manipulate elements of the group, combine them and analyze the resulting expressions.

One interpretation of these accounts with two non-negative values is as a trick to avoid using negative numbers (Ellerman 1985). We could have equivalently used negative numbers for credits and positive numbers for debits (or vice versa).³ The relationship between the real numbers and the α -elements is in fact one-to-one, there is one real number for every possible value of an account, as we show in Lemma 2. This can help the interpretation of these accounts considerably, as it allows us to use normal addition and subtraction on real numbers to understand the operations in this group.

Lemma 2. *There is a bijection from the group formed by $\{\mathcal{A}, +_{dc}\}$ to the group formed by $\{\mathcal{R}, +\}$, the real line and the group-law of addition.*

Proof. Proof is in the appendix. □

Through the equivalence relationship, we can organize all α elements based on the size of their debit or credit balance. If we make a line starting at 0 and increase the debit balances to the left, while increasing credit balances to the right, we get the horizontal axis in Figure 1. The figure then plots the real line orthogonal to this to show that for every possible value of an α -element, we find exactly 1 value on the real line (see the

³Historically, the relationship is likely the other way round. This system of notation was developed before the use of negative numbers took root. Since one can work with negative quantities of debit value, namely credit value, in this system without having to define negative numbers, it was a very convenient step in this development.

red line). The bijection –or one-to-one correspondence– between the α elements and the real line, is from every possible real number to all α -elements with the same value (an equivalence class). So the number -10 on the real line is mapped to an element holding the value 10 credit, regardless of whether that 10 credit is caused by crediting the element by 10 and debiting it by 0, or by crediting it by 10,005 and debiting it by 9995, or any other set of transactions that yields the same net balance. All of the elements that have 10 more credit than debit, have the same credit value of 10 and are therefore equivalent in this group definition. Also note that if we have an account with a credit value of 10 and we credit it by 10, we get a credit value of 20. The equivalent operation on the real line is taking -10 -10=-20. The mapping between the real line and the α accounts maps the value -20 to an account [0//20], so that indeed the operations have the same result. If we ever get confused about what we are doing when we process a journal entry in bookkeeping, we can therefore simply try to figure out what normal addition would do. Processing a journal entry in double-entry bookkeeping is mathematically equivalent to adding two sets of real numbers, add the credits to the credits and add the debits to the debits and the result should be correct.

Accounting systems have many accounts in which we keep track of different values. To describe these systems, we need a list of $n < \infty$ accounts, each with an individual α -element. Denote such lists with capital letters, such that $A = \{\alpha_1, \dots, \alpha_n\} \in \mathcal{A}^n$, while \mathcal{A}^n is the set of all lists of n α -elements. The subscript under the α -elements can be used to keep the different values that we are tracking separated. For instance, α_1 could be the value of our cash account, α_2 the value of an inventory, α_3 the value of the equity account, etc.

Ledgers, financial statements and journal entries all consists of lists of accounts. In the appendix I show that if we associate each account with a unique α -element, the resulting lists, A form a group with the same operator, now applied per index. This means the operator is applied to the debits and/or credits itemwise for all accounts. Figure 2 shows this operation graphically for a simple $A \in \mathcal{A}^4$. It shows how we can from a prototypical accounting system.

If we update the ledger in Figure 2a through the journal entry in Figure 2b, we have to debit account 1 and credit account 2 by the amounts from the journal entry. The resulting updated ledger-accounts, Figure 2c, again have both debit and credit values and only have non-negative numbers, so are also accounts from the set \mathcal{A} in this mathematical definition. Note also that the journal entry has two identity elements, the two zero-value elements in a_3, a_4 . As expected, the updated ledger shows they do not affect the value of the accounts a_3, a_4 . This also means that we can, and usually do, suppress their notation and only show the accounts that have a non-zero value.

As with the individual accounts, we call two lists equivalent if each of their elements has the same value and thus $\alpha_i \simeq \alpha'_i$ for all i in these lists. The group satisfies closure and associativity in the same way as each of the elements in the lists does. The identity element is a list of elements with zero value, and the inverse element of any list is the list where the debit and credit values are swapped within each α -element of the list. A similar bijection as the one that holds for a single α -element, holds for the entire list. So that the group $\{\mathcal{A}^n, +_{dc}\}$ has a one-to-one mapping to $\{\mathcal{R}^n, +\}$. This mapping can again be used to more easily understand the mathematical structure of the bookkeeping system, it is like adding and subtracting n -vectors of real numbers.

(a) Ledger	(b) Journal entry	(c) Updated ledger
$a_1 [x_1 // 0]$	$a_1 [0 // y_1]$	$a_1 [x_1 // y_1]$
$a_2 [x_2 // 0]$	$a_2 [y_2 // 0]$	$a_2 [x_2 + y_2 // 0]$
$a_3 [x_3 // 0]$	$a_3 [0 // 0]$	$a_3 [x_3 // 0]$
$a_4 [0 // x_4]$	$a_4 [0 // 0]$	$a_4 [0 // x_4]$
$a_5 [0 // x_5]$	$a_5 [0 // 0]$	$a_5 [0 // x_5]$

Index, i	Account name
1	Cash
2	Materials Inventory
3	Completed goods Inventory
4	Owner's equity
5	Loan, 5%
...	

(d) Association α -elements and accounts

Figure 2. A prototypical bookkeeping system.

The rules of bookkeeping have an additional important constraint, namely that any valid financial statement is balanced: *Debits = Credits*. For the balance sheet in Figure 2 to be balanced, we require that the sum of all debit values in all of the n accounts equals the sum of all credit values in the same list. Define the value of a list A as: $Val(A) = \sum_1^n d_i - \sum_1^n c_i$ where d_i, c_i denote the debit and credit values of the elements indexed by i . Similarly, let $dr(A)$, and $cr(A)$ denote the sum of all debit and credit values in the list A , respectively. Note that the value of an individual element (a list of length 1) is exactly equal to its net debit value, so this value is the mapping from \mathcal{A} to \mathcal{R} used in Lemma 2 and Figure 1. Our balance constraint thus means that from all the possible $A \in \mathcal{A}^n$, only those lists for which we can say that $Val(A) = dr(A) - cr(A) = 0$ are valid (well-formed) statements. Define this set, the set of all balanced lists of length n , as $\mathcal{B}^n = \{A \in \mathcal{A}^n | Val(A) = 0\}$. Denote each item in the list (i.e. an α -element from a balanced list) as b_i , where i is the same index as before.

By construction $\mathcal{B}^n \subset \mathcal{A}^n$, the elements in \mathcal{B} are all in \mathcal{A} , but the requirement that the lists in \mathcal{B}^n are balanced means that all non-balanced lists of length n that are in \mathcal{A}^n are not part of this smaller set. Even though we have a subset of the original group-set, this subset, together with the debit-credit addition, still forms a group:

Lemma 3. $\{\mathcal{B}, +_{dc}\}$, the set \mathcal{B} with the group-law of elementwise addition of debit and credit values forms a group.

As before the group definition is verified in the appendix.

A list B is a reasonable description of a lot of accounting entities. If we take a general ledger, for instance, it contains a number of accounts in which we keep track of different values. Each account has a different value and name, so they are separated and we could link each different account to a different index i (for instance via the decimal schedule)

in the list. To each account we can post both debit and credit values in a transaction, and we know that the sum of all credit values equals the sum of all debit values in the list, such that the general ledger is balanced.

With this notation and description of the balanced accounting group, we can derive the main proposition. This proposition shows how the constraint on the entire accounting system, that if we add up all the debit values in the list it equals the total of the credit values in the list, implies that every entry made, for all individual transactions, has to be balanced:

Proposition 1. *The difference between two balanced lists of accounts is balanced:*

$$\forall B, B' \in \mathcal{B} \quad (B -_{dc} B') \simeq B'' \in \mathcal{B}^n .$$

Proof. Proof is in the appendix. □

Mathematically, the result is straightforward. From the group definition we know inverses exist as elements of the group, thus inverses are balanced. Since adding the inverse is the same as taking difference under the addition operator, we are adding two balanced accounts to get a third balanced account. From an accounting point of view, the result is extremely relevant. The general ledger should always be balanced, so always be an element of \mathcal{B} . Now take the ledger at two different moments, moment 1 and 2, denoted with B^1, B^2 . Then we know that $B^2 - B^1 = \Delta B \in \mathcal{B}$. The difference between the general ledger at two different points in time is caused by journal entries that are processed. So the journal entries have to be equal to some $\Delta B \in \mathcal{B}$, and thus journal entries have to be an element of \mathcal{B} . We require the general ledger to be balanced and [Proposition 1](#) shows that this implies that all journal entries have to be balanced.

With a small addition, the same result also holds the other way round. If we start with a balanced set of account and require any change to be balanced, the resulting set of accounts are also balanced.

Corollary 1. *If some list of accounts(B_0) is balanced and all manipulations done to it (B_i) are also balanced, the resulting list of accounts is balanced:*

$$B_0, B_i \in \mathcal{B} \rightarrow B_0 +_{dc} B_1 +_{dc} \dots +_{dc} B_t = B^T \in \mathcal{B} \quad \forall 0 < T < \infty$$

Where B^T is used to denote the resulting element after T manipulations.

Proof. Since we know that $Val(B_i) = 0 \forall i$, it is immediate that $Val(B^T) = 0+0+..+0 = 0$. □

Before any transactions occurred, a company starts from a balanced list of zero values (i.e. before investment by owners in the company, the company does not have any value). If we restrict all later transactions to be recorded via balanced entries, the resulting general ledger is always balanced. These two restrictions, a balance of the general ledger, or a balance of every journal entry, are mathematically the same. This also means that we can check balance of the entire general ledger at the level of individual transactions. By checking that all journal entries are balanced when we make them, we make sure that the balance of the entire ledger is preserved.

Note that the result is much more general than just applying to the general ledger. The result holds for any balanced set of elements. So it holds, for instance, for any set of

accounts in our accounting system for which we know they hold balanced values. At the moment we make a balance sheet, we make sure the list of accounts it contains is balanced. All non-balance sheet items, that is all cost and revenue accounts that we keep during the period, are set to zero by adding any remaining value as a gain or loss on the equity account. The result in [Proposition 1](#) also applies to the difference between two balance sheets, regardless of whether this difference occurs over time, between economic entities, or between two different sets of valuation rules for the same entity at the same moment.

[Proposition 1](#) has several important consequences that are derived as corollaries: 1, unbalanced sets of accounts are easily detected and invalid by definition; 2, it implies the (expanded) accounting equations used in accounting courses; 3, it implies we can fill out missing pieces of information or identify redundant information in balanced sets of accounts.

2.1 Trial balance

Corollary 2. *A list of accounts B for which $\sum_i^{dc} B \neq [0 // 0]$ is not in \mathcal{B} , and thus not a valid set of financial statements. A list B' for which $\sum_i^{dc} B' \simeq [0 // 0]$ is in \mathcal{B} , and thus a valid set of financial statements.*

Since the balance constraint is part of the definition of the elements in the \mathcal{B} -group, it is immediate that non-balanced lists are not in \mathcal{B} . We use this property when we make a trial-balance to see if the ledger is still balanced. We record changes to the balance sheet between the start of the year and the end of the year through the journal entries and T-accounts in the ledger. For the trial balance we sum the balances in the T-accounts and see whether the total is balanced. Since we start with balanced books at the begin of the year, we know that if all entries were balanced, the resulting ledger should be balanced. This procedure thus checks that the changes that occurred are an element of \mathcal{B} .

We also use it for individual Balance Sheet accounts. Because we record all in- and out-flows of value to the stocks in our bookkeeping system, we know that the starting balance of any account on the balance sheet plus the sum of all inflows, equals the sum of outflows plus ending balance of the same account. Hence, the list of accounts denoting only the single balance sheet account and its in- and outflows also forms a balanced set of accounts, and thus follows the rules of the \mathcal{B} -group.

2.2 Accounting equations

The second corollary to [Proposition 1](#) derives another well-known accounting concept, the accounting equation.

Corollary 3. *[Proposition 1](#) implies that the changes in debit-balance accounts equal changes in credit-balance accounts.*

Again the result is immediate from [Proposition 1](#), the balance sheet itself is an element of \mathcal{B} , the changes over time to the balance sheet are thus elements of \mathcal{B} according to the proposition. The most well know application of this corollary is probably the use of the expanded accounting equation, an important part of introductory accounting courses, e.g. Weygandt et al. (2010, ch. 2), or Scott (2018, ch. 4):

$$assets_{t=0} + \Delta assets_1 = equity_{t=0} + \Delta equity_1 + liabilities_{t=0} + \Delta liabilities_1$$

Where the subscripts $t = 0$ denotes the start of period and the subscript 1 during the period. Note that by definition the assets at start of period equal the liabilities plus equity at start of period, so that we actually can break this down in:

$$\begin{aligned} assets_{t=0} &= equity_{t=0} + liabilities_{t=0} \\ \Delta assets_1 &= \Delta equity_1 + \Delta liabilities_1 \end{aligned}$$

Where the last line is a direct application of Corollary 3 and the first line is direct consequence of the definition of the β -accounts and owners' equity.

Note again that the result is more general; it does not just hold for the balance sheet but for any set of balanced accounts and the entire accounting system as a whole.⁴

2.3 Missing and redundant information

Proposition 1 implies that the accounting system has redundant information. There are numbers that we can leave out, without losing information. We can use these redundancies to fill out missing information, or remove the redundant information to gain more efficiency.

Corollary 4. *Proposition 1 implies that the \mathcal{B} -lists contains redundancies that allow identifying missing information.*

Proof. Split the list of accounts into a list A_C for which the information is complete and a missing list A_M , such that $A_C \cup A_M = B \in \mathcal{B}$. We then know that $Val(A_C) + val(A_M) = 0 \rightarrow val(A_C) = -val(A_M)$ and thus the debit balance of the known accounts is equal to the credit balance of the missing accounts. \square

Again this result should be familiar, as it is commonly used in relation to the balance sheet to fill out the last account. We can also (have students) use it to determine earnings by taking two balance sheets 1 year apart, denote them $A^1, A^2 \in \mathcal{B}$. By definition, we have $(A^1 - A^2) \in \mathcal{B}$ and thus the sum of the changes on the Debit side of the balance sheet is equal to the sum of changes on the Credit side. Then by summing all changes in all accounts except for the change in equity (denoted as A^2/α_e^2 where subscript e is for equity and $/$ is used to denote the set-exclusion operator) we get:

$$\begin{aligned} \sum_i^n \Delta val(\alpha_i) &= \sum_{i/e}^n (val(\alpha_i^2) - val(\alpha_i^1)) + val(\alpha_e^2) - val(\alpha_e^1) = 0 \\ val(A^2/\alpha_e^2) - val(A^1/\alpha_e^1) &= -(val(\alpha_e^2) - val(\alpha_e^1)) \end{aligned}$$

We get the equality that $val(A^2/\alpha_e^2) - val(A^1/\alpha_e^1) = val(\alpha_e^1) - val(\alpha_e^2)$. Thus, if we have the changes in assets and the changes in liabilities, we can find the change in equity as a direct application of this result. This equality is one possible reason that the

⁴This result is quite convenient. Yours truly is definitely guilty of explaining the accounting equation based on a simple balance sheet and then, without further explanation, applying the result to the entire accounting system, or inventories, or specific parts of the system that students happen to be examining. Any of these results only holds true because we can rely on Proposition 1.

bookkeeping system, as we know it, is balanced. The Venetian merchant-bankers who used the predecessor to our current bookkeeping invested heavily in companies outside of Venice, for instance, in Brussels and Antwerp. When they wanted to check the state of their investments, they would ask their agent to send over a copy of the books. As long as the books reflected all changes in possessions (assets) and debts, whatever remained was the money the agent owed to the merchant-banker, which is now known as owner's equity.

The result is again more general than helping to fill out the last remaining balance-sheet item, as it works for all balanced sets of accounts where only 1 account is missing (change the index e to any other account). This corollary also works the other way round. If we have some set of balanced accounts, B_1 , then we could remove one account without losing information (place it on the missing list A_M). We can then use the balance property to infer its value from the remaining numbers.

2.4 Double-entry bookkeeping: Accruals, matching, debits and credits and the balance constraint.

Another consequence of studying this mathematical structure of bookkeeping is that we can demonstrate that several prominent aspects of the modern bookkeeping system are not necessarily interconnected. We can have balanced books, trial balances, and the extra verifiability of the balance constraint, without ever using matching, debits and credits, or even accruals. In the last case, we should note, however, that unless we fundamentally want to revert to counting coins and kilograms of wheat, we will have to continue using accruals. They make keeping the system balanced a lot easier, since they allow us to move value at intermediate steps in production rather than only at purchase and sale.

From [Lemma 2](#) we already know we can map the accounts to the real line. After the mapping to the real line, the sum of all amounts equals zero. This one-to-one relationship also means that all results that we can prove for the \mathcal{B} -group must equivalently hold for the equivalent system built from numbers on the real line, as long as they form a list that adds up to zero. [Lemma 2](#) thus implies that the debits and credits are not necessary for our bookkeeping system to be balanced, they do allow us to present both inventory value and equity and liabilities as positive numbers, rather than force us to use negative numbers for one of the two.

Similarly, matching is not necessary for balanced books. When we match costs to a period (by an adjusting entry, for instance), we do so using a balanced journal entry. So we apply a balanced element to a balanced set of books. Just writing off an entire investment at purchase (like an R&D investment in the current system) or at disposal would not present a problem for balance. Similarly, if our value definition does not recognize revenue from sales on account as revenue, but recognizes it as some other (temporary) store of value, that would keep the accounts balanced.

When these observations are put together, they indicate a difficulty in the definition of double-entry bookkeeping: what properties does a system of bookkeeping need to have to be considered double-entry? The standard definition in accounting is that the whole system, put together, defines what we know as (scientific) double-entry bookkeeping. However, the balance constraint is not restricted to systems with debit and credit values, matching is not mathematically necessary for balance, nor does matching imply double

entry. There is even no need to use accruals (but without accruals we are back to counting coins). This makes the standard definition of double-entry bookkeeping a definition of an accounting system, encompassing questions of valuation, rather than just record-keeping.

3 discussion

In this paper I analyze the properties of an Abelian group that describes the algebra of a double-entry bookkeeping process with a balance constraint, like our current system. This shows the balance constraint forms the common origin for some well-known procedures used in bookkeeping, namely the effectiveness of a trial balance to check for errors, the use of the accounting equations, and the fact that one can use the information in the balance sheet to calculate missing values. These results hold in any recording system that enforces a balance between sets of accounts, whether each account has a debit and credit value, or is simply a single numeric value (that could go negative).

In our bookkeeping system, we keep track of all the stocks and all the flows of value in the company. Given that we do not allow for value to spontaneously appear or disappear, this is a closed system. Whatever value is created or lost, therefore has to be absorbed by or taken from the residual account, a role taken by the equity account. This definition of owner's equity, as the residual value of the assets in the company appears a likely reason for the the balance constraint in the development of the current accounting system (Renes Mimeo).

As section 2.4 shows, matching and debits and credits are not required for a balance constraint. However, it seems impossible to have a complete, closed system without this constraint. If we look at the formal construction of the \mathcal{B} -group, we can build a system that is equivalent in terms of the coded information to our current system, but does not have debits and credits in the way in the way it stores information. Furthermore, the loss in flexibility in such a balanced systems is relatively limited. It does not stop anyone from reading out individual accounts and using that information in later calculations. It also does not prevent more efficient storing of information in single, rather than double entries (i.e. a switch from double to single entry in the mathematical sense).

The balance requirement creates several types of verifiability in the accounting system. In the 'olden days' when bookkeeping was done by hand and errors in addition were common, the ability to verify your work in a trial balance was very worthwhile. With the advent of computers, this type of verifiability has become less valuable, computers make fewer errors in simple mathematics. However, the same constraint also limits the possibilities of manipulating the accounts by forcing the source of value to be recorded. The balance constraint creates several ways to measure the same information allowing us to check the processing of information in other parts of the system and, thus, can be part of a system to protect against intentional manipulations. If accounting systems will be extended beyond the single (corporate) entity to report over the entire value chain, these extra defenses are likely to become more valuable in the future.

References

- JGA Cripps. Particularis de computis et scripturis: 1494, luca pacioli; a contemporary interpretation. *Seattle: Pacoili Society*, 1995.
- David Ellerman. On double-entry bookkeeping: The mathematical treatment. *Accounting Education*, 23(5):483–501, 2014.
- David P Ellerman. The mathematics of double entry bookkeeping. *Mathematics Magazine*, 58(4):226–233, 1985.
- Katarina Gvozdic and Emmanuel Sander. Learning to be an opportunistic word problem solver: going beyond informal solving strategies. *ZDM*, pages 1–13, 2019.
- Luca Pacioli. *Summa de arithmetica, geometria, proportioni et proportionalita*. Paganino de paganini, Venice, 1494.
- Sander Renes. When debit=credit, the balance constraint in bookkeeping, its relevance for accounting valuation and the axiomatic design choices that cause it. *Working paper*, Mimeo.
- Alan Sangster. Pacioli’s lens: God, humanism, euclid, and the rhetoric of double entry. *The Accounting Review*, 93(2):299–314, 2017.
- Peter Scott. *Introduction to Accounting*. Oxford University Press, 1st edition edition, 2018.
- Jerry J Weygandt, Paul D Kimmel, and Donald Kieso. *Accounting principles*. Princeton University Press, international student version, 12th edition edition, 2010.

A \mathcal{A} groups, Lemma 1

In this section I show that the accounts meet the definition of an Abelian (symmetric) group under the operation of addition of the debit and credit values as is stated in [Lemma 1](#). The underlying set, \mathcal{A} , consists of the set of tuples of non-negative integers. The group law, or operation, is addition of the credit and debit values of the two accounts being operated on. To do so, first recall that we define an equivalence between two accounts $\alpha_1 = [a // b]$ and $\alpha_2 = [c // d]$ denoted as $\alpha_1 \simeq \alpha_2$, if we have that $a - b = c - d$ where a, b, c, d are non-negative integers by construction.

A.1 Closure and symmetry

If we have 2 accounts $\alpha_1 = [a // b]$ and $\alpha_2 = [c // d]$, the outcome of operation of addition is always the same: $\alpha_2 +_{dc} \alpha_1 \simeq \alpha_1 +_{dc} \alpha_2 \simeq [a + c // b + d] \simeq \alpha_3$ since if $0 \geq a, b, c, d < \infty$ then $0 \geq a + c, b + d < \infty$ and thus α_3 is finite and the outcome is again an account with two non-negative integers and thus in \mathcal{A} . So that the set is closed under this operation.

A.2 Associativity

Associativity requires that $(\alpha_1 +_{dc} \alpha_2) +_{dc} \alpha_3 \simeq \alpha_1 +_{dc} (\alpha_2 +_{dc} \alpha_3)$, given that we are dealing with addition, the order of addition does not impact the results, such that the operation is indeed associative.

A.3 Identity element

There is a unique identity element, $[0 // 0]$. By definition of the addition operation, we have $\alpha +_{dc} [0 // 0] \simeq \alpha$. Uniqueness is given by the fact that under the equivalence relationship we know that if

$$\alpha +_{dc} [x // x] \simeq \alpha +_{dc} [0 // 0] \tag{A.1}$$

$$[x // x] \simeq [0 // 0] \tag{A.2}$$

$$[x - x // 0] \simeq [0 // 0] \tag{A.3}$$

Where the last equivalence indeed holds since $x - x = 0$ by definition. Hence any element e that can be applied to another element a to yield a is equivalent to the identity element, $[0 // 0]$.

A.4 Inverse element

The inverse element maps the element to the identity element via the group law. Hence we need to guarantee that for each element in the set we can find an inverse element (denoted with superscript -1 , such that we have $\alpha^{-1} + \alpha = 0$). Via the equivalence relationship defined by the balance of the accounts, this element is easily identified for each tuple $[x // y]$ as $[y // x]$. We use this to denote the operation $+_{dc}\alpha_i^{-1}$ as $-\alpha_i$.

Since this means the set \mathcal{A} with the operation addition of the debit and credit values, satisfies the group axioms for closure, associativity, has a unique identity and inverse element (under the difference equivalence relation), so that it forms a group.

Posting journal entries to the ledger is symmetric. If we have 2 accounts $\alpha_1 = [a // b]$ and $\alpha_2 = [c // d]$, it does not matter whether we say that α_1 is the original account and α_2 comes from the journal entry or the other way round, the outcome of the application is always the same: $\alpha_2 +_{dc} \alpha_1 \simeq \alpha_1 +_{dc} \alpha_2 \simeq [a + c // b + d] \simeq \alpha_3$ and the outcome is again an account with two non-negative integers, so the group is an Abelian group.

B Isomorphism

The proof of the isomorphism to the real line is. Through the equivalence relationship, we can organize all α elements based on the size of their debit or credit balance. If we then make a line starting at 0 and increasing the credit balances to the left and increasing debit balances to the right, we get a line of elements in either direction. We can put the real line orthogonal to this as is done in [Figure 1](#), then for every possible value of an α -element we find exactly 1 value on the real line as is shown with the red line in [Figure 1](#). Furthermore, the relationship between the sets is one-to-one and can be defined for every element in both sets. The group law on the α -elements corresponds to normal addition on the real line, the inverse corresponds to the normal subtraction operation,

and the identity element maps to 0 the identity element on the real-line for addition. The equivalence relationship on the α -elements is then identical to the normal interpretation =.

C The list of accounts, **Lemma 3**

To prove **Lemma 3**, we have to prove the same properties for the lists of accounts that form the entire accounting system.

Associativity is given by the fact that \mathcal{B} elements consist of a list of \mathcal{A} -elements and each α -group is associative, such that the \mathcal{B} -elements consists of associative elements. Since we only apply elements with the same index to each other, associativity is given by the fact that each of the elements are associative.

Similarly the identity element can be obtained by combining n identity elements from the \mathcal{A} -group: $[0 // 0], \dots, [0 // 0]$.

The inverse element to $\beta = [\alpha_1, \dots, \alpha_n]$ is $\beta^{-1} = [\alpha_1^{-1}, \dots, \alpha_n^{-1}]$, which is unique as all of the elements have a unique inverse.

This group is also an Abelian group, as we have to do the addition per account, and thus have the same result per account as we had under the \mathcal{A} -group.

Any finite number of addition applications of accounts to each other, or any finite number of journal entries applied to some set of accounts yields a set of transformed accounts $B' \in \mathcal{R}^n$ by the same logic. That is, we can conclude that the set of elements $B \in \mathcal{R}^n$ is closed under finite debit-credit addition. While for each element in \mathcal{B} we know that $\sum_i^{dc} B \simeq [0 // 0]$, so that $\sum_i B +_{dc} \sum_i B' \simeq [0 // 0] +_{dc} [0 // 0] \quad \forall B, B' \in \mathcal{B}$. Or in bookkeeping terms, as long as we process valid journal entries, we end up with a valid set of financial statements for any finite number of journal entries. Which proofs **Lemma 3**.

D Differences are balanced

to prove that the difference between two balanced lists of accounts is balanced:

$$\forall B, B' \in \mathcal{B} \quad (B -_{dc} B') \simeq B'' \in \mathcal{B}^n .$$

We first need some notation. Technically $B -_{dc} B'$ should be read as $B +_{dc} (B')^{-1}$ where $(B')^{-1}$ is the inverse element of B' . This is equivalent to the definition of subtraction over the real numbers, so I will use the minus sign for ease of notation. By definition of the inverse elements, the credit-balance of B is the debit-balance of $-B$. By associativity of the group, the order of addition does not matter, so we can write $val(B -_{dc} B') = val(B) + val(B'^{-1}) = 0 - 0 = 0$, so the difference between to balanced lists is similarly balanced.