The Unreasonable Effectiveness of Bitstrings in Logical Geometry

Hans Smessaert and Lorenz Demey

Abstract. This paper presents a unified account of *bitstrings* — i.e. sequences of bits (0/1) that serve as compact semantic representations — for the analysis of Aristotelian relations and provides an overview of their effectiveness in three key areas of the Logical Geometry research programme. As for *logical* effectiveness, bitstrings allow a precise and positive characterisation of the notion of logical independence or unconnectedness, as well as a straightforward computation — in terms of bitstring length and level — of the number and type of Aristotelian relations that a particular formula may enter into. As for *diagrammatic* effectiveness, bitstrings play a crucial role in studying the subdiagrams of the Aristotelian rhombic dodecahedron, and different types of Aristotelian hexagons turn out to require bitstrings of different lengths. The *linguistic* and *cognitive* effectiveness of bitstring analysis relates to the scalar structure underlying the bitstrings, and to the difference between linear and non-linear bitstrings.

Mathematics Subject Classification (2000). Primary 03G05, 68T30; Secondary 03B65, 06A07, 00A66, 97C30.

Keywords. Bitstrings, Logical Geometry, Aristotelian diagram, logical effectiveness, unconnectedness, diagrammatic effectiveness, linguistic effectiveness, cognitive effectiveness.

1. Introduction

The central aim of the research programme of *Logical Geometry* (henceforth abbreviated as LG) is to develop an interdisciplinary framework for the study of logical diagrams.[1] LG has focussed on constructing logical diagrams for (i) *logical* systems such as syllogistics with subject negation [10], syllogistics with singular propositions [41], modal logic [40] and public announcement logic [9], (ii) *linguistic*

[1]For more detailed information, see the website `www.logicalgeometry.org`.

systems such as those involving subjective quantifiers [46] and generalised quantifiers [40], and (iii) *conceptual* systems, such as those involving the Aristotelian and duality relations themselves [14] and the metalogical concepts of tautology and satisfiability [11].

LG studies both the abstract-logical properties and the visual-geometrical properties of logical diagrams. As far as the *abstract-logical* properties of logical diagrams are concerned, LG investigates a range of topics including the information contents of the Aristotelian relations [43], the difference between opposition and implication relations [43], the intricate connection between Aristotelian and duality relations [17, 42], the context-dependence of Aristotelian relations [10], logical complementarities between Aristotelian diagrams [44, 45, 46], Boolean subfamilies and Boolean closures of Aristotelian diagrams [41]. These abstract-logical topics are studied from the perspective of logic itself [10], but also from those of formal semantics [40, 42], group theory [8, 17] and lattice theory [13, 40].

As for the *visual-geometrical* properties of logical diagrams, the LG framework studies, among others, the relation between Aristotelian and Hasse diagrams [13, 40], differences between 2D and 3D diagrams [40, 46], subdiagrams embedded inside larger diagrams [8, 41, 44, 45, 46], geometrical complementarities between Aristotelian diagrams [44, 45, 46], informational and computational equivalence of Aristotelian diagrams [15, 16, 47] and cognitive aspects of Aristotelian and duality diagrams [8, 13]. For the analysis of these visual-geometrical topics LG makes crucial use of insights from disciplines such as cognitive psychology [13, 44], group theory [8, 17], diagrams design [13, 44] and computer graphics [10].[2]

The LG programme also studies the historical development of logical diagrams, focussing on their use in the works of distinguished authors such as John Buridan [19] and J. N. Keynes [10]. Finally, LG has also explored the potential roles of logical diagrams in logic education [11] and the interface between formal and natural languages [14].

In its investigations, LG makes extensive use of *bitstrings*, i.e. sequences of bits (0/1) that serve as compact representations of the formulas' semantics. These bitstrings have turned out to be an extremely powerful tool, yielding both quantitative and qualitative results as well as raising interesting new questions. The main aims of this paper are hence (1) to present a unified account of bitstrings in LG and (2) to provide an overview of their effectiveness in the various areas of LG.[3]

The paper is organised as follows. Section 2 introduces bitstrings and discusses some of their basic properties. The next three sections survey the effectiveness of bitstrings in three key areas of LG. In particular, Section 3 goes into the logical effectiveness of bitstrings, while Section 4 deals with their diagrammatic

[2]Some of these abstract-logical and visual-geometrical properties are also studied (for Aristotelian diagrams) in Moretti's oppositional geometry framework [28, 29].

[3]The paper thus stands in a long tradition of work discussing the 'unreasonable effectiveness' of a variety of mathematical tools and techniques for a variety of purposes [4, 21, 22], which was initiated by Wigner's famous [49].

effectiveness, and Section 5 addresses their linguistic and cognitive effectiveness. Finally, Section 6 draws some conclusions and points out some prospects and challenges.

2. Bitstrings in Logical Geometry

Bitstrings are sequences of bits (0/1) which serve as compact combinatorial representations, both of the denotations of formulas in *logical systems* (such as classical propositional logic, first-order logic, modal logic and public announcement logic), and of concepts from *lexical fields* (such as comparative quantification, subjective quantification, color terms and set inclusion relations).[4] As such, there is no limitation on the length of bitstrings: they may consist of any number of bit positions. For example, bitstrings consisting of up to 16 bit positions have already proved useful in LG [10]. However, most of the properties and applications to be discussed in this paper can already be described by means of much shorter bitstrings. For ease of presentation, we will therefore mainly work with bitstrings of length 4, which allow us to encode various interesting logical fragments, such as the $2^4 = 16$ formulas of classical propositional logic with 2 propositional variables p and q, and the 16 formulas from the modal logic $\mathsf{S5}$ with 1 propositional variable p, as illustrated in Table 1. If a formula φ is encoded by the bitstring b, we write $\beta(\varphi) = b$. In other words, β is a function mapping a formula φ onto its bitstring b.[5] Bitstrings can be characterised in terms of their *level*, i.e. the number of positions with value 1. Hence, for bitstrings of length 4, the top half in Table 1 contains the 4 level 1 (L1) bitstrings 1000, 0100, 0010 and 0001 and their 4 contradictory L3 bitstrings 0111, 1011, 1101 and 1110. The bottom half in Table 1 then consists of the 6 L2 bitstrings as well as the L0 and L4 bitstrings 0000 and 1111.

The Aristotelian relations are standardly defined as relations holding between two *formulas*. Relative to a logical system S (which is assumed to be bivalent, and have all the Boolean connectives), two formulas φ, ψ are said to be

S-*contradictory* (CD_{S})	iff	$\mathsf{S} \models \neg(\varphi \wedge \psi)$	and	$\mathsf{S} \models \neg(\neg\varphi \wedge \neg\psi)$,
S-*contrary* (C_{S})	iff	$\mathsf{S} \models \neg(\varphi \wedge \psi)$	and	$\mathsf{S} \not\models \neg(\neg\varphi \wedge \neg\psi)$,
S-*subcontrary* (SC_{S})	iff	$\mathsf{S} \not\models \neg(\varphi \wedge \psi)$	and	$\mathsf{S} \models \neg(\neg\varphi \wedge \neg\psi)$,
in S-*subalternation* (SA_{S})	iff	$\mathsf{S} \models \varphi \rightarrow \psi$	and	$\mathsf{S} \not\models \psi \rightarrow \varphi$.

[4]The original formulation of bitstring semantics in Smessaert [40] was inspired by considerations from generalised quantifier theory about partitioning the powerset of the quantificational domain. As demonstrated in Chatti [5, 6], however, an informal precursor of this technique was already used by Avicenna in the 11th century AD. Conceptually very similar techniques are the *setting* approach of Pellissier [30], the *valuation spaces* account of Seuren [37, 39] and the *question-answer semantics* of Schang [33].

[5]Note that Moretti [29] and Schang [34] use a bitstring-like notation to encode the Aristotelian relations themselves (as well as possible generalisations of these relations). Within the LG framework, however, bitstrings do not encode *relations* between formulas, but rather (the denotations of) the *formulas* as such. Finally, note that it is not always immediately clear how to define the bitstring mapping β precisely; however, a systematic way for achieving this is available (also see Section 6 and [18]).

Table 1. Bitstrings (BS) for the 16 formulas of classical propositional logic (CPL) and the modal logic S5.

S5	CPL	BS	BS	CPL	S5
$\Box p$	$p \wedge q$	1000	0111	$\neg(p \wedge q)$	$\neg\Box p$
$p \wedge \neg\Box p$	$\neg(p \rightarrow q)$	0100	1011	$p \rightarrow q$	$\neg p \vee \Box p$
$\neg p \wedge \Diamond p$	$\neg(p \leftarrow q)$	0010	1101	$p \leftarrow q$	$p \vee \neg\Diamond p$
$\neg\Diamond p$	$\neg(p \vee q)$	0001	1110	$p \vee q$	$\Diamond p$
S5	CPL	BS	BS	CPL	S5
p	p	1100	0011	$\neg p$	$\neg p$
$\Box p \vee (\neg p \wedge \Diamond p)$	q	1010	0101	$\neg q$	$\neg\Box p \wedge (p \vee \neg\Diamond p)$
$\Box p \vee \neg\Diamond p$	$p \leftrightarrow q$	1001	0110	$\neg(p \leftrightarrow q)$	$\neg\Box p \wedge \Diamond p$
$\Box p \wedge \neg\Box p$	$p \wedge \neg p$	0000	1111	$p \vee \neg p$	$\Box p \vee \neg\Box p$

This definition shows that the Aristotelian relations are sensitive with respect to the logical system S [10, 18]. If the system is clear from the context, we will usually omit it, and simply talk about 'contrariety' instead of 'S-contrariety', and so on. As will be discussed in more detail in Section 3, this definition is fundamentally 'hybrid' in nature: the relations *CD*, *C* and *SC* are defined in terms of whether the formulas can be true together and whether they can be false together,[6] whereas *SA* is defined in terms of implication or truth propagation [43].

Completely analogously, the Aristotelian relations can be defined as holding between two *bitstrings*. Two bitstrings b_1 and b_2 of length ℓ are said to be

contradictory (CD)	iff	$b_1 \wedge b_2 = 0 \cdots 0$	and	$b_1 \vee b_2 = 1 \cdots 1$,
contrary (C)	iff	$b_1 \wedge b_2 = 0 \cdots 0$	and	$b_1 \vee b_2 \neq 1 \cdots 1$,
subcontrary (SC)	iff	$b_1 \wedge b_2 \neq 0 \cdots 0$	and	$b_1 \vee b_2 = 1 \cdots 1$,
in subalternation (SA)	iff	$b_1 \wedge b_2 = b_1$	and	$b_1 \vee b_2 \neq b_1$.

If two formulas φ and ψ cannot be true together, the meet of the corresponding bitstrings $\beta(\varphi)$ and $\beta(\psi)$ equals the bottom element of the Boolean algebra $\{0,1\}^\ell$, namely the L0 bitstring $0 \cdots 0$.[7] Similarly, φ and ψ cannot be false together, whenever the join of the bitstrings $\beta(\varphi)$ and $\beta(\psi)$ equals the top element of the Boolean algebra $\{0,1\}^\ell$, namely the Lℓ bitstring $1 \cdots 1$. The Aristotelian relation holding between any two formulas can then easily be determined by computing the meet and join of their bitstring counterparts. In other words, the formulas φ and ψ

[6]The $\neg(\varphi \wedge \psi)$ part in these definitions specifies whether the formulas can be true together, while the $\neg(\neg\varphi \wedge \neg\psi)$ part specifies whether the formulas can be false together. Note that these clauses explicitly use the $\neg$-connective to express that a formula is false, and thus assume the classicality of the underlying logical system S. In non-classical (e.g. many-valued) logics, the informal condition that two formulas cannot be true (resp. false) together can be formalised in many different, non-equivalent ways.

[7]The Boolean operations on bitstrings are defined bitwise, i.e. as operations of negation, conjunction or disjunction computed bit position by bit position. For example, $\neg 1100 = 0011$, $1100 \wedge 1010 = 1000$ and $1100 \vee 1010 = 1110$.

stand in some Aristotelian relation (as defined for S) if and only if $\beta(\varphi)$ and $\beta(\psi)$ stand in that same relation (as defined for bitstrings). This can be seen as a manifestation of the representation theorem for finite Boolean algebras [20, chapter 15].

In contrast to the setting approach of Pellissier [30], the mapping β assigns a *semantics* to the formulas. More in particular, each bit provides an answer to a meaningful (binary) question. In the case of $\mathsf{S5}$, for instance, the bit positions encode answers to the following questions about sets of possible worlds (PWs), where φ is a modal formula containing the propositional variable p:

Is φ true if p is true in all PWs?	yes/no
Is φ true if p is true in the actual world but not in all PWs?	yes/no
Is φ true if p is true in some PWs but not in the actual world?	yes/no
Is φ true if p is true in no PWs?	yes/no

The examples below illustrate how the bitstrings of length 4 that the β-function assigns to the formulas of $\mathsf{S5}$ are a compact way to represent a quadruple of yes/no answers to the questions above:

$$\begin{array}{lclcl} \beta(\Diamond p) & = & 1110 & = & \langle \text{ yes, yes, yes, no } \rangle \\ \beta(\Diamond p \wedge \Diamond\neg p) & = & 0110 & = & \langle \text{ no, yes, yes, no } \rangle \\ \beta(\Diamond\neg p) & = & 0111 & = & \langle \text{ no, yes, yes, yes } \rangle \end{array}$$

The fact that the $\mathsf{S5}$-formula in the middle example is the conjunction of the upper and lower formulas nicely corresponds to its bitstring being the meet of the upper and lower bitstrings as well as to its quadruple of answers being the meet of the upper and and lower quadruples.

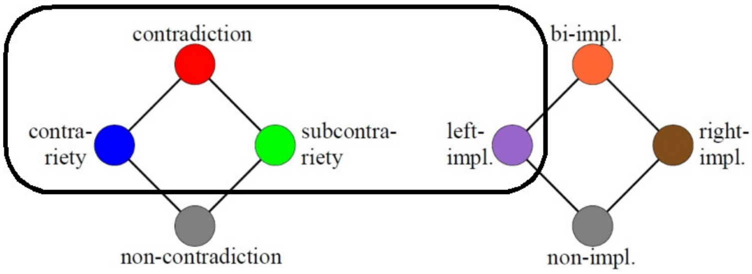
3. Logical Effectiveness

This section discusses two prime examples of the logical effectiveness of bitstring semantics. First of all, bitstrings allow us to provide a precise and positive characterisation of the notion of logical independence or unconnectedness. Secondly, the number and type of Aristotelian relations that a particular formula may enter into can straightforwardly be computed on the basis of the length and the level of its bitstring representation.

3.1. Characterizing Unconnectedness

As was mentioned in the previous section, the original set of Aristotelian relations is hybrid. In [43] two other sets of logical relations are defined in order to account for this hybrid nature, namely the *opposition* relations and the *implication* relations. The set of opposition relations is uniformely defined in terms of whether the formulas can be true together and whether they can be false together, and is obtained by removing subalternation from the original set of Aristotelian relations and replacing it with the relation of non-contradiction:

FIGURE 1. Aristotelian relations as hybrid between opposition and implication relations.



Opposition relations between bitstrings. Two bitstrings b_1 and b_2 of length ℓ are

contradictory (CD)	iff	$b_1 \wedge b_2 = 0 \cdots 0$	and	$b_1 \vee b_2 = 1 \cdots 1$,
contrary (C)	iff	$b_1 \wedge b_2 = 0 \cdots 0$	and	$b_1 \vee b_2 \neq 1 \cdots 1$,
subcontrary (SC)	iff	$b_1 \wedge b_2 \neq 0 \cdots 0$	and	$b_1 \vee b_2 = 1 \cdots 1$,
non-contradictory (NCD)	iff	$b_1 \wedge b_2 \neq 0 \cdots 0$	and	$b_1 \vee b_2 \neq 1 \cdots 1$.

The set of implication relations, by contrast, is uniformely defined in terms of implication or truth propagation. The starting point is the relation of subalternation which was removed from the Aristotelian relations and relabeled as left-implication since the implication holds from the first/left formula to the second/right formula (but not vice versa). The three extra implication relations then correspond to implication from right to left (right-implication), two-way implication (bi-implication), and absence of implication in either direction (non-implication):

Implication relations between bitstrings. Two bitstrings b_1 and b_2 of length ℓ are in

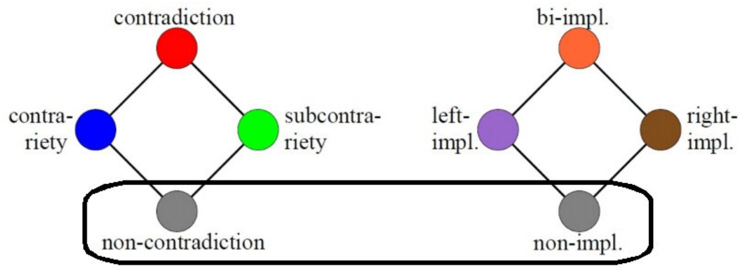
bi-implication (BI)	iff	$b_1 \wedge b_2 = b_1$	and	$b_1 \vee b_2 = b_1$,
left-implication (LI)	iff	$b_1 \wedge b_2 = b_1$	and	$b_1 \vee b_2 \neq b_1$,
right-implication (RI)	iff	$b_1 \wedge b_2 \neq b_1$	and	$b_1 \vee b_2 = b_1$,
non-implication (NI)	iff	$b_1 \wedge b_2 \neq b_1$	and	$b_1 \vee b_2 \neq b_1$.

In Figure 1 the hybrid nature of the Aristotelian relations is visualised: the relations of contradiction, contrariety, and subcontrariety are taken from the set of opposition relations on the left, whereas subalternation corresponds to left-implication from the set of implication relations on the right.[8]

In [43] the lattices for the two sets of relations in Figure 1 are argued to be ordered by information level: they reveal parallel hierarchies of informativity, with the least informative relations at the bottom, and the most informative ones at

[8]Despite their conceptual independence, there are several close connections between the sets of opposition and implication relations — e.g. $LI(b_1, b_2)$ iff $C(b_1, \neg b_2)$ [43, Lemma 3]. The latter essentially captures Schang's [35] claim that subalterns can be seen as contradictories of contraries; also see [43, Footnote 18].

Figure 2. Unconnectedness as the combination of non-contradiction and non-implication.



the top. From an informational perspective, the four Aristotelian relations can be considered maximally informative.[9]

This information perspective also sheds new light on the notion of unconnectedness. Classically, two formulas are said to be *unconnected* if and only if they do not stand in any Aristotelian relation whatsoever.[10] As illustrated in Figure 2, the information perspective provides an alternative, positive characterisation of unconnectedness in terms of the two least informative opposition and implication relations, viz. non-contradiction and non-implication respectively.

It can be shown that unconnectedness requires bitstrings of length at least 4: if two formulas φ and ψ are unconnected, then their bitstring representations $\beta(\varphi)$ and $\beta(\psi)$ need to consist of at least 4 bit positions. Since unconnectedness is defined as the combination of non-contradiction and non-implication, and the latter two themselves are both characterised in terms of two conditions, unconnectedness involves four conditions altogether. By virtue of non-contradiction, two unconnected formulas φ and ψ can be true together and can be false together. In terms of their bitstring representations, this means that there must be at least one bit position in which both $\beta(\varphi)$ and $\beta(\psi)$ have a value 1, and at least one bit position in which both $\beta(\varphi)$ and $\beta(\psi)$ have a value 0 respectively. By virtue of non-implication, there can be no implication relation in either direction between two unconnected formulas φ and ψ. In terms of their bitstring representations, this means that there must be at least one bit position in which $\beta(\varphi)$ has a value 1 and $\beta(\psi)$ has a value 0, and conversely, that there must be at least one bit position in which $\beta(\psi)$ has a value 1 and $\beta(\varphi)$ has a value 0. Since these four conditions on bit positions are logically independent, φ and ψ can only be unconnected if their bitstrings $\beta(\varphi)$ and $\beta(\psi)$ consist of at least 4 bit positions.[11] By contraposition,

---

[9] The absence of the two informative implication relations of bi-implication and right-implication can be accounted for independently, see [43].

[10] Many authors refer to this same notion as *logical independence*, e.g. see [1, 23, 31, 37].

[11] This dual perspective on unconnectedness can already be found in the works of the 14th century logician John Buridan. He characterised unconnected formulas negatively as "obeying no law, neither the law of contradictories, nor the law of contraries, nor the law of subcontraries, nor

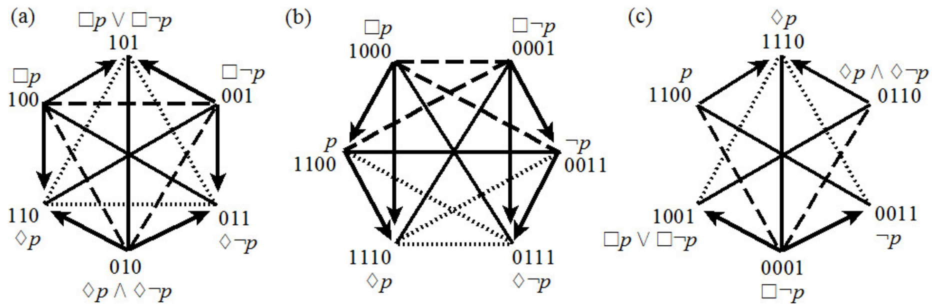


FIGURE 3. Three Aristotelian hexagons for S5: (a) strong Jacoby-Sesmat-Blanché, (b) Sherwood-Czeżowski, (c) unconnected-4.

it also holds that if the formulas in an Aristotelian diagram can be encoded by bitstrings of length 3, then that diagram cannot contain any unconnectedness, i.e. every pair of its formulas stands in some Aristotelian relation.

Consider the three examples of Aristotelian hexagons for S5 in Figure 3. The best-known hexagon is no doubt the strong Jacoby-Sesmat-Blanché (JSB) hexagon in Figure 3(a): it can be encoded by bitstrings of length 3, and thus does not contain any unconnectedness.[12] It is important to stress that having bitstrings of length 4 is a necessary condition for unconnectedness, but not a sufficient condition. In other words, it is perfectly possible to have Aristotelian diagrams that require an encoding by means of bitstrings of length at least 4, and that yet do not contain any unconnectedness. A case in point is the Sherwood-Czeżowski (SC) hexagon of Figure 3(b), which requires bitstrings of length 4, but in which every pair of formulas nevertheless does stand in some Aristotelian relation.[13] By contrast, the unconnected-4 (U4) hexagon in Figure 3(c) does contain unconnectedness (e.g. the formulas p and $\Diamond p \wedge \Diamond\neg p$ are unconnected), and therefore its formulas can only be encoded by bitstrings of length at least 4.[14]

that of subalterns”, whereas according to his positive characterisation, “such propositions can be true at the same time ... and they can both be false at the same time ... [and] it is impossible that one should follow from the other” [3, p. 81]. Also see [19].

[12]The strong JSB hexagon in Figure 3(a) is named after Jacoby [24], Sesmat [36] and Blanché [2].

[13]The SC hexagon in Figure 3(b) is named after William of Sherwood [25, 26] and Czeżowski [7].

[14]The U4 hexagon in Figure 3(c) is called ‘unconnected-4’ because it contains exactly 4 pairs of unconnected formulas; it has recently been studied in [38] and [44].

3.2. Calculating Logical Relations

A second illustration of the logical effectiveness of the bitstring approach concerns the way in which, for any bitstring of length ℓ and level i, we can use simple combinatorial arguments on bitstrings[15] to calculate the number of:

contradictories	$\#CD$	$= 1$
contraries	$\#C$	$= 2^{\ell-i} - 1$
subcontraries	$\#SC$	$= 2^i - 1$
non-contradictories	$\#NCD$	$= (2^{\ell-i} - 1)(2^i - 1)$

If we take a level 1 bitstring of length 3, for instance, then $\ell = 3$ and $i = 1$, which yields the following distribution over the 4 opposition relations:

$$\begin{array}{lll} \#CD & & = 1 \\ \#C & = 2^{\ell-i} - 1 = 2^{3-1} - 1 = 2^2 - 1 = 4 - 1 & = 3 \\ \#SC & = 2^i - 1 = 2^1 - 1 = 2 - 1 & = 1 \\ \#NCD & = (2^{\ell-i} - 1)(2^i - 1) = 3 \times 1 & = 3 \end{array}$$

Notice that the total number of relations equals $2^3 = 8$, since every bitstring of length 3 stands in an opposition relation to itself and to the 7 other bitstrings of length 3 (i.e. including the bottom element 000 and the top element 111). For example, for the L1 bitstring 100 the distribution looks as follows:[16]

$$\begin{array}{rl} CD[100] & = \{011\} \\ C[100] & = \{010, 001, 000\} \\ SC[100] & = \{111\} \\ NCD[100] & = \{110, 101, 100\} \end{array}$$

Completely analogously, taking a level 2 bitstring of length 4 ($\ell = 4$ and $i = 2$) yields the following distribution over the 4 opposition relations:

$$\begin{array}{lll} \#CD & & = 1 \\ \#C & = 2^{\ell-i} - 1 = 2^{4-2} - 1 = 2^2 - 1 = 4 - 1 & = 3 \\ \#SC & = 2^i - 1 = 2^2 - 1 = 4 - 1 & = 3 \\ \#NCD & = (2^{\ell-i} - 1)(2^i - 1) = 3 \times 3 & = 9 \end{array}$$

For the L2 bitstring 1100, for instance, the $2^4 = 16$ bitstrings are distributed over the opposition relations in the following manner:

$$\begin{array}{rl} CD[1100] & = \{0011\} \\ C[1100] & = \{0010, 0001, 0000\} \\ SC[1100] & = \{1011, 0111, 1111\} \\ NCD[1100] & = \{1000, 0100, 1010, 1001, 0110, 0101, 1100, 1110, 1101\} \end{array}$$

Finally, it can be shown that for bitstrings on non-extreme levels (i.e. which are on level i, for $1 < i < \ell - 1$), we have $\#CD < \#C, \#SC < \#NCD$. There thus exists a perfect inverse correlation between (i) the numbers of opposition relations that

---

[15]The combinatorial arguments for $\#CD$, $\#C$ and $\#SC$ can also be found in [35] (where they are based on Schang's question-answer semantics).

[16]For any binary relation R on a set A, the *R-image* of an element $a \in A$ is defined as $R[a] := \{a' \in A \mid (a, a') \in R\}$.

those bitstrings enter into, and (ii) the informativity ordering of the opposition relations shown in Figures 1 and 2:

$$\begin{array}{rccccc} \text{number of relations} & \#CD & < & \#C, \#SC & < & \#NCD \\ \text{informativity ordering} & CD & > & C, SC & > & NCD \end{array}$$

Notice, furthermore, that if $i \approx \frac{\ell}{2}$, then $\#C \approx \#SC$. In other words, bitstrings in middle levels have similar numbers of contraries and subcontraries, which straightforwardly corresponds to the fact that contrariety and subcontrariety occupy the same intermediate level of informativity in the lattices of Figures 1 and 2 [43].[17]

4. Diagrammatic Effectiveness

This section presents two key examples of the diagrammatic effectiveness of bitstring analysis. First, bitstrings play a crucial role in studying the subdiagrams of the Aristotelian rhombic dodecahedron. Second, in establishing an exhaustive typology of all possible Aristotelian hexagons, different types of hexagons turn out to require bitstrings of different lengths.

4.1. Subdiagrams of the Aristotelian Rhombic Dodecahedron

The JSB hexagon in Figure 3(a) is *Boolean closed*: every contingent Boolean combination of formulas in this hexagon is (logically equivalent to) a formula that already belongs to it. It thus visualises the entire Boolean algebra $\{0,1\}^3$, except for its $\top$-element 111 and $\bot$-element 000. The SC hexagon in Figure 3(b), by contrast, is *not* Boolean closed: the disjunction of the two top vertices, for instance, is itself not (logically equivalent to) a vertex of the hexagon. The construction of the Boolean closure of bitstrings of length 4 has led to the discovery of the rhombic dodecahedron (RDH) — a 3D polyhedron with 12 rhombic faces and 14 vertices — for the visualisation of the Boolean algebra $\{0,1\}^4$, represented by bitstrings of length 4 [40]. In order to describe the internal structure of this RDH and to present an exhaustive typology of all Aristotelian diagrams that can be embedded inside RDH, bitstrings again play a crucial role. The $2^4 - 2 = 14$ contingent bitstrings of $\{0,1\}^4$ constitute 7 pairs of contradictories (PCDs). These 7 PCDs can be subdivided into 4 C-PCDs — which correspond to the 4 diagonals of the *cube* embedded in RDH and connect the L1 and L3 bitstrings — and 3 O-PCDs — which correspond to the 3 diagonals of the *octahedron* embedded in RDH and connect pairs of L2 bitstrings. This so-called CO-perspective then yields an exhaustive typology of the subdiagrams of RDH in terms of how many C-PCDs and how many O-PCDs they consist of. For example, both the strong Jacoby-Sesmat-Blanché hexagon in Figure 3(a) and the Sherwood-Czeżowski hexagon in Figure 3(b) are C^2O^1 hexagons, whereas the unconnected-4 hexagon in Figure 3(c) is a C^1O^2 hexagon [45, 48].

[17] The application of combinatorial techniques to bitstrings has generated many more results that are relevant for LG than the few simple ones described in this subsection. A more comprehensive and mathematically detailed overview can be found in [12].

TABLE 2. Bitstring compression from length 4 to length 3.

1**01**1	1**10**1	1**00**1	$\rightsquigarrow$	1**0**1	1**11**0	$\rightsquigarrow$	1**1**0
1**01**0	1**10**0	1**00**0	$\rightsquigarrow$	1**0**0	0**11**1	$\rightsquigarrow$	0**1**1
0**01**1	0**10**1	0**00**1	$\rightsquigarrow$	0**0**1	0**11**0	$\rightsquigarrow$	0**1**0
0**01**0	0**10**0	(0**00**0)	$\rightsquigarrow$	(0**0**0)	(1**11**1)	$\rightsquigarrow$	(1**1**1)

As far as embedding smaller Aristotelian diagrams into bigger ones is concerned, the classical result in the literature is that the RDH contains six strong JSB hexagons [27, 28, 30, 32, 40, 46]. Bitstrings turn out to be a very powerful tool to study such embeddings. If we consider two bit positions, for example the second and third, then the 14 contingent bitstrings of length 4 can be partitioned into a group of 8 bitstrings having different values in those positions — the left-hand side of Table 2 — and a group of 6 bitstrings having identical values in those positions — the right-hand side of Table 2.[18] The latter group constitutes a strong JSB hexagon, whereas the former group constitutes its complementary Buridan octagon [44, 45, 46]. Although we are dealing with bitstrings of length 4, the six contingent bitstrings in the right half of Table 2 — with identical values in their second and third bit positions — can thus be 'compressed' into bitstrings of length 3, which constitute the JSB hexagon in Figure 3(a).[19]

There are exactly $\frac{4\times 3}{2} = 6$ ways in which bitstrings of length 4 can have identical (resp. different) values in two of their bit positions, and these correspond exactly to the 6 strong JSB hexagons (resp. Buridan octagons) embedded inside RDH:[20]

$[b]_2 = [b]_3$	$[b]_1 = [b]_2$	$[b]_3 = [b]_4$	$[b]_1 = [b]_4$	$[b]_1 = [b]_3$	$[b]_2 = [b]_4$
JSB1	JSB2	JSB3	JSB4	JSB5	JSB6

For the modal logic S5, the first three JSB hexagons are presented in terms of classical, paraconsistent and paracomplete negation in [1]. The fourth JSB hexagon was discovered independently in [28, 40] and the fifth and sixth JSB hexagons are introduced in [30, 40].[21]

We have just seen that the strong JSB hexagons inside RDH can be characterised by means of bitstring constraints of the form $[b]_i = [b]_j$ (for distinct $i, j \in \{1, 2, 3, 4\}$). It can be shown that all other types of Aristotelian diagrams embedded inside RDH can also be characterised by means of other, more complex

[18] Of course, the top and bottom elements 1111 and 0000 also have identical values in their second and third bit positions, but as usual, these are ignored in Aristotelian diagrams, which explains the numerical discrepancy between the two groups.

[19] For example, by collapsing the second and third bit positions, the bitstrings 1**00**0 and 0**11**0 for $\Box p$ and $\Diamond p \wedge \Diamond \neg p$ in RDH are compressed into the bitstrings 1**0**0 and 0**1**0 in Figure 3(a), respectively.

[20] We will write $[b]_i = [b]_j$ to express the condition that a bitstring b has the *same* values in bit positions i and j.

[21] Note that the corresponding six hexagons for CPL were already discovered in [32] and that [27] establishes the connection between S5 and CPL.

bitstring constraints. For example, SC hexagons are characterised by bitstring constraints of the form $[b]_i \neq [b]_j \wedge ([b]_i = [b]_k \rightarrow [b]_i = [b]_\ell)$ (for pairwise distinct $i, j, k, \ell \in \{1, 2, 3, 4\}$); the concrete SC hexagon shown in Figure 3(b) corresponds to taking $i = 1, j = 4, k = 3$ and $\ell = 2$.

4.2. An Exhaustive Typology of Aristotelian Hexagons

A second illustration of the diagrammatic effectiveness of bitstrings concerns the typology of Aristotelian hexagons. A first question to be answered is how many hexagons can be constructed with bitstrings of length ℓ. Although strictly speaking there are 2^ℓ bitstrings of length ℓ, the restriction to contingent bitstrings means we generally only consider $2^\ell - 2$ bitstrings of length ℓ. The following combinatorial formula captures the number of hexagons with bitstrings of length ℓ:

$$\frac{(2^\ell-2)(2^\ell-4)(2^\ell-6)}{3! \times 2^3}$$

Bitstrings are chosen in contradictory pairs (PCDs): choosing one bitstring automatically means choosing its contradictory as well. Hence, in order to select a hexagon, only three 'choices' need to be made in the numerator of this fraction, and the number of bitstrings from which we can choose each time decreases by 2 instead of 1. The denominator captures the variety of presentations of a given hexagon: 3! represents the number of permutations of 3 PCDs, while 2^3 reflects the fact that each of these 3 PCDs occurs inside the hexagon with a given 'orientation' (e.g. 1000—0111 versus 0111—1000).[22] Applying the formula above to bitstrings of length 3 to 7 yields the following numbers of hexagons:

$\ell = 3$	$\ell = 4$	$\ell = 5$	$\ell = 6$	$\ell = 7$
$\frac{6\times4\times2}{48}$	$\frac{14\times12\times10}{48}$	$\frac{30\times28\times26}{48}$	$\frac{62\times60\times58}{48}$	$\frac{126\times124\times122}{48}$
1	35	455	4495	39711

Secondly, bitstrings have proved their computational importance in generating all possible types of Aristotelian hexagons (and their Boolean subtypes). They thus allow us to answer the question which types of hexagons exist and which lengths of bitstrings each type requires. As discussed before, the strong JSB hexagon in Figure 3(a) requires bitstrings of length 3, whereas the Sherwood-Czeżowski and unconnected-4 hexagons in Figure 3(b-c) require bitstrings of length 4. Three other types of Aristotelian hexagons can be distinguished: the weak JSB hexagon [30] and the (strongest Boolean subtype of the) unconnected-12 hexagon also require bitstrings of length 4, whereas the (strongest Boolean subtype of the) unconnected-8 hexagon is the only type requiring bitstrings of length 5.[23] A combination of mathematical reasoning and exhaustive computational verification has

[22] See [16] for a detailed comparison of the relationship between the number of presentations of a hexagon on the one hand, and the number of geometrical symmetries/rotations of a regular hexagon on the other.

[23] From the bitstring characterisations of the strong and weak JSB hexagons, it follows that a JSB hexagon is strong iff it is Boolean closed.

demonstrated that there exist no types of Aristotelian hexagons that require length 6 or higher (up to Boolean subtype).

5. Linguistic and Cognitive Effectiveness

This section briefly introduces two topics illustrating the linguistic and cognitive effectiveness of bitstring analysis, namely that of the scalar structure underlying bitstrings, and that of the difference between linear and non-linear bitstrings.

5.1. Scalar Structure in Bitstrings

In addition to its logical and diagrammatical effectiveness, bitstring semantics also generates new questions about the linguistic and cognitive aspects of the encoded expressions. Two related questions are (i) what is the relative weight or strength of individual bit positions inside bitstrings? and (ii) what is the scalar or linear structure of the underlying conceptual domain? To illustrate these questions, note that the semantics of the basic operators of modal logic, predicate logic and total orders in Figure 4 can all straightforwardly be captured in terms of bitstrings of length 3. Nevertheless, there does seem to be a clear intuitive difference in the relative weight of the individual bit positions in these cases, in the sense that some bit positions correspond to *points* on a cognitive scalar structure (or 'logical space'), whereas other bit positions correspond to *intervals* on that structure. In the case of the modal operators and the quantifiers in Figure 4(a-b), for instance, the first and third bit position encode the end points of the scale, whereas the second bit position encodes the intervening interval. With the ordering relations in Figure 4(c), by contrast, the second bit position encodes the central reference point on the scale, whereas the first and third positions encode the intervals extending to the left and to the right of that reference point.

The tripartitions in Figure 4(a-b) can then be seen as the result of superimposing two bipartitions that each consist of one point and one interval, e.g. *all* vs. *not all* (with the point on the left and the interval on the right) and *some* vs. *no* (with the interval on the left and the point on the right). By contrast, the scalar structure of total orders in Figure 4(c) can either be seen as being primitively tripartite in nature, or alternatively as being the result of superimposing two bipartitions that each consist of two intervals, viz. $\geq$ vs. $<$ on the one hand, and $>$ vs. $\leq$ on the other (so that the central reference point of the tripartite scale ($=$) only arises out of the interaction between these two bipartitions).

It should be emphasised that the distinction between point- and interval-interpretations of bit positions is primarily relevant from a linguistic or cognitive perspective, and does not go beyond the realm of classical Boolean algebra. In particular, the scalar structures in Figure 4(a-c) all share the same Boolean structure. For example, for all three scalar structures, the negation of the middle bit position is identical to the join of the leftmost and rightmost bit positions ($\neg 010 = 101 = 100 \vee 001$), regardless of whether that middle bit position corresponds to an interval (as in Figure 4(a-b)) or to a point (as in Figure 4(c)).

FIGURE 4. Points versus intervals in bitstrings of length 3.

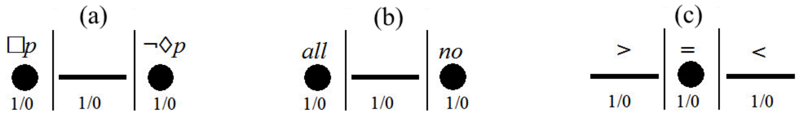
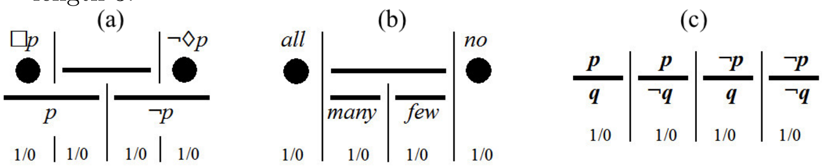


FIGURE 5. Bitstrings of length 4 as refinements of bitstrings of length 3.



Moving from bitstrings of length 3 to those of length 4, some quadripartite scalar structures can be seen as refinements of an underlying tripartite scalar structure, while others seem to be primitively quadripartite in nature, or to be the result of superimposing two bipartitions. For example, the quadripartite scale of the modal logic S5 in Figure 5(a) can be seen as the result of superimposing a bipartition for the bare modalities (p vs. $\neg p$) onto the original tripartition of Figure 4(a) ($\Box p$ vs. $\Diamond p \wedge \Diamond \neg p$ vs. $\neg \Diamond p$). Similarly, the bipartition with the subjective quantifiers *many* and *few* in Figure 5(b) can be seen as a further refinement of the original interval of the second bit position in Figure 4(b) [46]. With the formulas of CPL in Figure 5(c), by contrast, the scalar structure can either be seen as being primitively quadripartite in nature (with each bit position corresponding to a row in the classical truth tables), or alternatively as being the result of superimposing two independent bipartitions (viz. p vs. $\neg p$ and q vs. $\neg q$).

5.2. Linear versus Non-linear Bitstrings

From a mathematical or algebraic perspective we cannot distinguish between 'linear' bitstrings — such as 1010, where all four bit positions are linearly ordered with respect to each other — and 'non-linear' bitstrings — such as $1{}^{0}_{1}0$, where the precise ordering between the second and the third bit position is left unspecified. From a linguistic or cognitive perspective, however, such a difference does become relevant. *Linear* bitstrings imply that all questions — i.e. all bit positions — about a lexical field can be situated on a single dimension. For the realms of comparative and proportional quantification this does indeed seem to be the case. *Non-linear* bitstrings, by contrast, imply that the various questions belong to fundamentally distinct dimensions, as was argued to be the case for the modalities of S5 and the scale with *many* and *few* in Figure 5(a-b).

It should be emphasised that from a mathematical perspective, linear and non-linear bitstrings have the same Boolean structure. For example, a non-linear bitstring such as $1_1^0 0$ consists of *four* bit positions that each have *exactly one* of the values 1 and 0 (just like the linear bitstring 1010). In particular, the non-linear bitstring $1_1^0 0$ should not be seen as consisting of *three* bit positions, with the second position containing *both* the values 1 and 0. The latter perspective might also prove useful (e.g. for assigning bitstring semantics to non-classical logics), but by allowing certain bit positions to be simultaneously 1 and 0, it constitutes a far more radical departure from the realm of classical Boolean algebra than the non-linear bitstrings proposed here.

In future research, empirical hypotheses will be formulated concerning the cognitive complexity of various lexical fields (e.g. in terms of processing times), and possible correlations with the scalar and (non-)linear nature of their underlying bitstring representations will be investigated.

6. Conclusion

In this paper we have presented a unified account of bitstrings and provided an overview of their effectiveness in three key areas of the Logical Geometry research programme. As for *logical* effectiveness, bitstrings first of all allow us to provide a positive characterisation of the notion of unconnectedness as the combination of two conditions for non-contradiction and two conditions for non-implication, thus requiring bitstrings of length at least 4. Secondly, the number and type of Aristotelian relations that a particular formula may enter into can straightforwardly be computed on the basis of the length and the level of its bitstring representation. The number of opposition relations ($\#CD < \#C, \#SC < \#NCD$) turns out to be inversely correlated with the informativity level of these relations.

Furthermore, two key examples have been discussed regarding the *diagrammatic* effectiveness of bitstring semantics. On the one hand, bitstrings play a crucial role in studying the subdiagrams of the Aristotelian rhombic dodecahedron. A case in point is the embedding of 6 strong JSB hexagons in RDH, which can be accounted for in terms of the 6 ways in which a bitstring of length 4 can be compressed into a bitstring of length 3 by collapsing bit positions with identical values. On the other hand, the exhaustive typology of all possible Aristotelian hexagons reveals that different types of hexagons require bitstrings of different lengths. Four types require a bitstring length of 4 (the weak JSB, the Sherwood-Czeżowski, the Unconnected-4 and the Unconnected-12 hexagons), whereas the strong JSB hexagon only requires length 3 and the Unconnected-8 hexagon requires length 5.

Finally, two topics have briefly illustrated the *linguistic* and *cognitive* effectiveness of bitstring analysis. First of all, scalar structures underlying the bitstrings may differ from one another as to which bit positions correspond to points on the scale and which positions to intervals. Some quadripartite scalar structures can be considered as refinements of originally tripartite structures, whereas others are inherently quadripartite. Secondly, bitstrings are called linear or non-linear depending on whether the underlying binary questions relate to one single dimension or to different dimensions.

As illustrated throughout the paper, bitstrings have proved extremely useful in Logical Geometry so far. Nevertheless, bitstring analysis in its original formulation (as presented in this paper) still exhibits a number of limitations. First of all, it is not always clear how 'sensitive' bitstrings are to the specific properties of the underlying logical system: two formulas may enter into different Aristotelian relations with one another depending on the logical system and should therefore be assigned different bitstrings accordingly. Secondly, the complex interplay between Boolean and Aristotelian structure requires further investigation: some fragments which have an isomorphic Aristotelian structure may nevertheless not be isomorphic from a Boolean point of view. Thirdly, the current approach does not provide a systematic strategy for establishing a bitstring semantics for any fragment $\mathcal{F}$ of any logical system S [9]. In ongoing research we are developing a more mathematically mature version of bitstring semantics that is able to overcome these different limitations [18].

Acknowledgments

We thank Dany Jaspers, Alessio Moretti, Fabien Schang and Margaux Smets for their comments on earlier versions of this paper. The second author holds a post-doctoral fellowship from the Research Foundation–Flanders (FWO).

References

[1] J.-Y. Béziau, *New Light on the Square of Oppositions and its Nameless Corner.* Logical Investigations **10** (2003), 218–232.

[2] R. Blanché, *Structures Intellectuelles. Essai sur l'Organisation Systématique des Concepts.* Vrin, 1969.

[3] J. Buridan, *Summulae de Dialectica.* Translated by Gyula Klima, Yale University Press, 2001.

[4] S. Burr (ed.), *The Unreasonable Effectiveness of Number Theory.* American Mathematical Society, 1992.

[5] S. Chatti, *Logical Oppositions in Arabic logic: Avicenna and Averroes.*, In J.-Y. Béziau (ed.) Around and Beyond the Square of Opposition, Springer (2012), 21–40.

[6] S. Chatti, *Avicenna on Possibility and Necessity.* History and Philosophy of Logic **35** (2014), 332–353.

[7] T. Czeżowski, *On Certain Peculiarities of Singular Propositions.* Mind **64** (1955), 392–395.

[8] L. Demey, *Algebraic Aspects of Duality Diagrams.* In P. Cox, B. Plimmer and P. Rodgers (eds.), Diagrammatic Representation and Inference, Lecture Notes in Computer Science 7352, Springer (2012), 300–302.

[9] L. Demey, *Structures of Oppositions for Public Announcement Logic.* In J.-Y. Béziau and D. Jacquette (eds.), Around and Beyond the Square of Opposition, Springer (2012), 313–339.

[10] L. Demey, *Interactively Illustrating the Context-Sensitivity of Aristotelian Diagrams.* In H. Christiansen, I. Stojanovic and G. Papadopoulos (eds.), Modeling and Using Context, Lecture Notes in Computer Science 9405, Springer (2015), 331–345.

[11] L. Demey, *Using Syllogistics to Teach Metalogic.* Submitted manuscript (2016).

[12] L. Demey, *Combinatorial Considerations on Logical Geometry.* Manuscript (2016).

[13] L. Demey and H. Smessaert, *The Relationship between Aristotelian and Hasse Diagrams.* In T. Dwyer, H. Purchase and A. Delaney (eds.), Diagrammatic Representation and Inference, Lecture Notes in Computer Science 8578, Springer (2014), 213–227.

[14] L. Demey and H. Smessaert, *Logische Geometrie en Pragmatiek.* In F. Van de Velde, H. Smessaert, F. Van Eynde and S. Verbrugge (eds.), Patroon en argument, Leuven University Press (2014), 553–564.

[15] L. Demey and H. Smessaert, *Shape Heuristics in Aristotelian Diagrams.* In O. Kutz, S. Borgo and M. Bhatt (eds.), Shapes 3.0 – The Shape of Things, CEUR-WS (2016).

[16] L. Demey and H. Smessaert, *The Interaction between Logic and Geometry in Aristotelian Diagrams.* In M. Jamnik, Y. Uesaka and S. Schwarz (eds.), Diagramatic Representation and Reasoning, Lecture Notes in Computer Science 9781, Springer (2016).

[17] L. Demey and H. Smessaert, *Duality in Logic and Language.* In J.-Y. Béziau (ed.), Encyclopedia of Logic, College Publications (forthcoming).

[18] L. Demey and H. Smessaert, *Combinatorial Bitstring Semantics for Arbitrary Logical Fragments.* Submitted manuscript (2016).

[19] L. Demey and P. Steinkrüger, *De logische geometrie van Johannes Buridanus' modale achthoek [The Logical Geometry of John Buridan's Modal Octagon].* Tijdschrift voor Filosofie (forthcoming).

[20] S. Givant and P. Halmos, *Introduction to Boolean Algebras.* Springer, 2009.

[21] J. Halpern, R. Harper, N. Immerman, P. Kolaitis, M. Vardi and V. Vianu, *On the Unusual Effectiveness of Logic in Computer Science.* Bulletin of Symbolic Logic **7** (2001), 213–236.

[22] R. Hamming, *The Unreasonable Effectiveness of Mathematics.* American Mathematical Monthly **87** (1980), 81–90.

[23] G. Hughes, *The Modal Logic of John Buridan.* In G. Corsi, C. Mangione en M. Mugnai (eds.), Atti del convegno internzionale di storia della logica, le teorie delle modalità, CLUEB (1987), 93–111.

[24] P. Jacoby, *A Triangle of Opposites for Types of Propositions in Aristotelian Logic.* The New Scholasticism **24** (1950), 32–56.

[25] Y. Khomskii, *William of Sherwood, singular propositions and the hexagon of opposition.* In J.-Y. Béziau and G. Payette (eds.), The Square of Opposition. A General Framework for Cognition, Peter Lang (2012), 43–60.

[26] N. Kretzmann, *William of Sherwood's Introduction to Logic.* Minnesota Archive Editions, 1966.

[27] D. Luzeaux, J. Sallantin and C. Dartnell, *Logical Extensions of Aristotle's Square.* Logica Universalis **2** (2008), 167–187

[28] A. Moretti, *The Geometry of Logical Opposition.* PhD thesis, University of Neuchâtel, 2009.

[29] A. Moretti, *From the "Logical Square" to the "Logical Poly-Simplexes". A Quick Survey of what Happened in Between.* In J.-Y. Béziau and G. Payette (eds.), The Square of Opposition. A General Framework for Cognition, Peter Lang (2012), 119-156.

[30] R. Pellissier, *"Setting" n-Opposition.* Logica Universalis **2** (2008), 235–263.

[31] S. Read, *John Buridan's Theory of Consequence and his Octagons of Opposition.* In J.-Y. Béziau and D. Jacquette (eds.), Around and Beyond the Square of Opposition, Springer/Birkhäuser (2012), 93–110.

[32] P. Sauriol, *Remarques sur la Théorie de l'Hexagone Logique de Blanché.* Dialogue **7** (1968), 374–390.

[33] F. Schang, *Abstract Logic of Oppositions.* Logic and Logical Philosophy **21** (2012), 415–438.

[34] F. Schang, *Questions and Answers about Oppositions.* In J.-Y. Béziau and G. Payette (eds.), The Square of Opposition. A General Framework for Cognition, Peter Lang (2012), 289-320.

[35] F. Schang, *Logic in Opposition.* Studia Humana **2** (2013), 31–45.

[36] A. Sesmat, *Logique II. Les Raisonnements.* Hermann, 1951.

[37] P. Seuren, *The Logic of Language. Language from Within, volume II.* Oxford University Press, 2010.

[38] P. Seuren, *From Whorf to Montague. Explorations in the Theory of Language.* Oxford University Press, 2013.

[39] P. Seuren, *The Cognitive Ontogenesis of Predicate Logic.* Notre Dame of Journal of Formal Logic **55** (2014), 499-532.

[40] H. Smessaert, *On the 3D Visualisation of Logical Relations.* Logica Universalis **3** (2009), 303–332.

[41] H. Smessaert, *Boolean Differences between Two Hexagonal Extensions of the Logical Square of Oppositions.* In P. Cox, B. Plimmer and P. Rodgers (eds.), Diagrammatic Representation and Inference, Lecture Notes in Computer Science 7352, Springer (2012), 193–199.

[42] H. Smessaert, *The Classical Aristotelian Hexagon Versus the Modern Duality Hexagon.* Logica Universalis **6** (2012), 171–199.

[43] H. Smessaert and L. Demey, *Logical Geometries and Information in the Square of Opposition.* Journal of Logic, Language and Information **23** (2014), 527–565.

[44] H. Smessaert and L. Demey, *Logical and Geometrical Complementarities between Aristotelian Diagrams.* In T. Dwyer, H. Purchase and A. Delaney (eds.), Diagrammatic Representation and Inference, Lecture Notes in Computer Science 8578, Springer (2014), 246-260.

[45] H. Smessaert and L. Demey, *La Géometrie Logique du Dodécaèdre Rhombique des Oppositions.* In: Chatti, S. (ed.) Le Carré et ses Extensions: Approches Théoriques, Pratiques et Historiques, Université de Tunis (2015), 127–157.

[46] H. Smessaert and L. Demey, *Béziau's Contributions to the Logical Geometry of Modalities and Quantifiers.* In A. Koslow and A. Buchsbaum (eds.), The Road to Universal Logic, Volume I, Springer (2015), 475–494.

[47] H. Smessaert and L. Demey, *Visualising the Boolean Algebra* $\mathbb{B}_4$ *in 3D.* In M. Jamnik, Y. Uesaka and S. Schwarz (eds.), Diagramatic Representation and Reasoning, Lecture Notes in Computer Science 9781, Springer (2016).

[48] H. Smessaert and L. Demey, *The Logical Geometry of the Aristotelian Rhombic Dodecahedron.* Manuscript (2016).

[49] E. Wigner, *The Unreasonable Effectiveness of Mathematics in the Natural Sciences.* Communications on Pure and Applied Mathematics **13** (1960), 1–14.

Hans Smessaert
Research Group on Formal and Computational Linguistics
KU Leuven
Blijde Inkomststraat 21
3000 Leuven, Belgium
e-mail: `hans.smessaert@kuleuven.be`

Lorenz Demey
Center for Logic and Analytic Philosophy
KU Leuven
Kardinaal Mercierplein 2
3000 Leuven, Belgium
e-mail: `lorenz.demey@kuleuven.be`