On the "mysterious" effectiveness of mathematics in science

J Gerard Wolff*

May 19, 2017

Abstract

This paper notes first that the effectiveness of mathematics in science appears to some writers to be "mysterious" or "unreasonable". Then reasons are given for thinking that science is, at root, the search for compression in the world. At more length, several reasons are given for believing that mathematics is, fundamentally, a set of techniques for compressing information and their application. From there, it is argued that the effectiveness of mathematics in science is because it provides a means of achieving the compression of information which lies at the heart of science. The anthropic principle provides an explanation of why we find the world—aspects of it at least—to be compressible.

Information compression may be seen to be important in both science and mathematics, not only as a means of representing knowledge succinctly, but as a basis for scientific and mathematical inferences—because of the intimate relation that is known to exist between information compression and concepts of prediction and probability.

The idea that mathematics may be seen to be largely about the compression of information is in keeping with the view, supported by evidence that is outlined in the paper, that much of human learning, perception, and cognition may be understood as information compression. That connection is itself in keeping with the observation that mathematics is the product of human ingenuity and an aid to human thinking.

Keywords: information compression, philosophy of mathematics, philosophy of science.

*Dr Gerry Wolff, BA (Cantab), PhD (Wales), CEng, MBCS (CITP); CognitionResearch.org, Menai Bridge, UK; jgw@cognitionresearch.org; +44 (0) 1248 712962; +44 (0) 7746 290775; *Skype*: gerry.wolff; *Web*: www.cognitionresearch.org.

1 Introduction

Although mathematics is a phenomenally successful "handmaiden" of science,[1] the reason that it is so effective in science has been described as a "mystery" that is "unreasonable". Thus:

- Roger Penrose writes:

> "It is remarkable that *all* the SUPERB theories of Nature have proved to be extraordinarily fertile as sources of mathematical ideas. There is **a deep and beautiful mystery** in this fact: that these superbly accurate theories are also extraordinarily fruitful simply as *mathematics*." ([Penrose, 1989, pp. 225–226], bold face added).

- In a similar vein, John Barrow writes:

> "**For some mysterious reason** mathematics has proved itself a reliable guide to the world in which we live and of which we are a part. Mathematics works: as a result we have been tempted to equate understanding of the world with its mathematical encapsulization. ... **Why is the world found to be so unerringly mathematical?**" ([Barrow, 1992, Preface, p. vii], bold face added).

- And Eugene Wigner [1960] writes about "The unreasonable effectiveness of mathematics in the natural sciences":

> "The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which **we neither understand** nor deserve. We should be grateful for it and hope that it will remain valid in future research and that it will extend, for better or for worse, to our pleasure, even though perhaps also to **our bafflement**, to wide branches of learning." (*ibid*, p. 14, bold face added).

In the light of evidence and arguments described in sections that follow, there appears to be a solution to the mystery of why mathematics is so effective in science. That solution is described in Sections 14 and 15.

[1] The slightly whimsical idea that mathematics might be some kind of servant of science, and the use of the curiously archaic word "handmaiden" seems to have originated with *The Handmaiden of the Sciences*, a book by Eric Temple [1937].

In brief: 1) Apart from the gathering of empirical data, science may be seen to be essentially a process of compressing those data; 2) mathematics may be seen to be a set of techniques for compression of information and their application; thus 3) For those reasons, mathematics can be a valuable aid in the process of compressing information which is a central part of good science. 4) The anthropic principle appears to explain why we find that much of the world is compressible (Section 15).

Information compression may be seen to be important in both science and mathematics, not only as a means of representing knowledge succinctly, but as a basis for scientific and mathematical inferences—because of the intimate relation that is known to exist between information compression and concepts of prediction and probability (Appendix D).

2 Human learning, perception, and thinking as information compression

It is pertinent to mention that much of the thinking in this paper derives from the development of the *SP theory of intelligence* and its realisation in the *computer model*, introduced in Appendix A.

A central idea in the SP system is that much of human learning, perception, and thinking may be understood as information compression. Although this may seem implausible, there is now much supporting evidence. Some of this evidence is relatively direct, described in Wolff [1993], [Wolff, 2006, Chapter 2], and Wolff [2017]. Less direct but nevertheless strong evidence is the way in which the SP computer model, which is dedicated to the compression of information, can model several different aspects of intelligence. Much of this evidence is presented in Wolff [2006]; Wolff, 2016] with pointers to where further information may be found.

If it is accepted that much of human cognition may be understood as compression of information, then it should not be surprising to find that both science and mathematics, as products of the human intellect, may also be understood as compression of information.

3 Science as compression of information

Occam's Razor, the principle attributed to William of Ockham and widely seen as a key principle in science, is often expressed as "Entities are not to be multiplied beyond necessity."—meaning that, when there are two or more competing theories that explain a given set of phenomena, we should choose the simplest.

Of course, much of science is concerned with observational studies of aspects of the "world"—meaning the universe as far as we can see—or conducting experiments to obtain empirical data. But few would dispute the 'elegance' or 'beauty' of a compact expression like $E = mc^2$ compared with the relatively huge range of observations that it describes or predicts, and few would dispute the importance in science of discovering or inventing compact descriptions like that.

Respected scientists have often described the goals of science in similar terms. Isaac Newton wrote that "Nature is pleased with simplicity" [Newton, 2014, p. 320]; Ernst Mach [2004] and Karl Pearson [1892] suggested independently of each other that scientific laws promote "economy of thought"; Albert Einstein wrote that "A theory is more impressive the greater the simplicity of its premises, the more different things it relates, and the more expanded its area of application.";[2] and cosmologist John Barrow has written that "Science is, at root, just the search for compression in the world" [Barrow, 1992, p. 247]. It is pertinent to mention that George Kingsley Zipf developed the related idea that human behaviour is governed by a "principle of least effort" [Zipf, 1949].

Here are some examples of simplifications in science:

> "... as space and time fuse together in a single concept of spacetime, so the electric field and the magnetic fields fuse together in the same way, merging into a single entity which today we call the electromagnetic field. The complicated equations written by Maxwell for the two fields become simple when written in this new language. ... The concepts of 'energy' and 'mass' become combined in the same way as time and space, and electric and magnetic fields, are fused together in the new mechanics. ... Einstein realizes that energy and mass are two facets of the same entity, just as the electric and magnetic fields are two facets of the same field, and as space and time are two facets of the one thing: spacetime. This implies that mass, by itself, is not conserved; and energy—as it was conceived at the time—is not independently conserved either. One may be transformed into the other: only one single law of conservation exists, not two. What is conserved is the sum of mass and energy, not each separately. Processes must exist that transform energy into mass, or mass into energy." [Rovelli, 2016, Location 812].

3.1 Simplicity and power

Since competing theories rarely address exactly the same set of phenomena, Occam's Razor may be adapted to be "In the development of a scientific theory, we

[2] Quoted in Isaacson [2007, p. 512].

should try to maximise the *simplicity* of the theory whilst retaining as much as possible of its descriptive or explanatory *power*."

There is a close connection between Occam's Razor as just described and the concept of compressing a body of information, **I**. This may be seen to be a process of maximising the *simplicity* of **I**, by extracting repeated information or *redundancy* from **I**, whilst retaining as much as possible of its non-redundant descriptive *power*.

A qualification here is that the results of information compression may be divided into two parts: a 'grammar' **G**, and an 'encoding' of **I** in terms of **G**, which we may call **E**. Here, **G** and **E** together represent lossless compression of **I**. However, **G** may be regarded as a 'theory' of **I** which is normally more 'interesting' than **E**. For reasons of that sort, **E** may sometimes be discarded (Appendix B).

3.2 Representation of knowledge and concepts of prediction and probability

There is much more to information compression than simply reducing the size of a body of information. As described in Appendix D, there is an intimate relation between information compression and concepts of prediction and probability.

Hence, compression of information is important in science, partly as a means of representing scientific knowledge in a succinct form—but at least as important is how information compression provides the key to the making of inferences and the calculation of probabilities.

4 Mathematics as compression of information

The second step in the argument, depends on evidence that mathematics is fundamentally about the compression of information, with a set of techniques for achieving that compression. This section and ones that follow present evidence in support of this idea, which we may refer to as mathematics-as-information-compression or MAIC.

5 An example of information compression via mathematics

It has been noted already how Einstein's equation, $E = mc^2$, may be seen to be a very compact representation of much data. Here is another example that demonstrates how ordinary mathematics—not some specialist algorithm for the compression of information—can yield high levels of information compression.

Newton's equation for his second law of motion, $s = (gt^2)/2$, is a very compact means of representing any realistically-large table showing the distance travelled by a falling object (s) in a given time since it started to fall (t), as illustrated in Table 1.[3] That small equation would represent the values in the table even if it was a 1000 times or a million times bigger, and so on. Likewise for other equations such as $a^2 + b^2 = c^2$, $PV = k$, $F = q(E + v \times B)$, and so on.

Distance (m)	*Time (sec)*
0.0	0
4.9	1
19.6	2
44.1	3
78.5	4
122.6	5
176.5	6
240.3	7
313.8	8
397.2	9
490.3	10
593.3	11
706.1	12
828.7	13
961.1	14
1103.2	15
1255.3	16
Etc	*Etc*

Table 1: The distance travelled by a falling object (metres) in a given time since it started to fall (seconds).

6 Some basic principles and techniques for information compression

What is it that makes it possible to represent large amounts of data with a very compact equation? This section outlines some basic principles and techniques that

[3]Of course, the law does not work for something like a feather falling in air. The constant, g, is the acceleration due to gravity—about $9.8m/s^2$.

may be seen to underpin most techniques for information compression, including those at work within in mathematics.

Perhaps the most fundamental principle is a very simple idea: that we may identify repetition or *redundancy* in information by searching for patterns that match each other, and that we may reduce that redundancy and thus compress information by merging or *unifying* two or more matching patterns to make one. This idea—*information compression via the matching and unification of patterns*—may be referred to in brief as "ICMUP".

6.1 Variants of ICMUP

There are five main variants of ICMUP, all of which are widely used in everyday life. The five variants are:

- *Chunking-with-codes.* With the first variant—a technique called *chunking-with-codes*—the unified pattern, often referred as a "chunk" of information, is given a relatively short name, identifier, or "code" which is used as a shorthand for the chunk of information wherever it occurs (except for a single 'master' copy). If, for example, the words "Treaty on the Functioning of the European Union" appear in several different places in a document, we may save space by writing the expression once, giving it a short name such as "TFEU", and then using that name as a code or shorthand for the expression wherever it occurs. Likewise for the abbreviation "ICMUP" that is used in this paper.

- *Schema-plus-correction.* Another variant, *schema-plus-correction*, is like chunking-with-codes but the unified chunk of information may have variations or "corrections" on different occasions. For example, a six-course menu in a restaurant may have the general form "`M1: Appetiser (S) sorbet (M) (P) coffee-and-mints`", with choices at the points marked '`S`' (starter), '`M`' (main course), and '`P`' (pudding). Then a particular meal may be encoded economically as something like '`M1:(3)(5)(1)`', where the digits determine the choices of starter, main course, and pudding.

- *Run-length coding.* A third variant, *run-length coding*, may be used where there is a sequence of two or more copies of a pattern, each one except the first following immediately after its predecessor. In this case, the multiple copies may be reduced to one, as before, with something to say how many copies there are, or when the sequence begins and ends, or, more vaguely, that the pattern is repeated without anything to say when the sequence stops. For example, a sports coach might specify exercises as something like

"touch toes (×15), push-ups (×10), skipping (×30), ..." or "Start running on the spot when I say 'start' and keep going until I say 'stop' ".

- *Class-inclusion hierarchies with inheritance of attributes.* In this variant, there is a hierarchy of classes and subclasses, with "attributes" at each level. Each attributes may be seen as a chunk of information and the corresponding class name may be seen to be its code. At every level except the top level, the subclass "inherits" the attributes of all higher levels.

- *Part-whole hierarchies with inheritance of contexts.* This is much the same as class-inclusion hierarchies with inheritance of attributes except that the structure represents the parts and subparts of some entity. In this case, each subpart may be seen to inherit its place in larger structures and thus the contexts of structures with which it is associated.

6.2 Hiding in plain sight

These techniques for information compression are so familiar that they are often "hiding in plain sight": widely used because they seem like the obvious way to express things, but rarely with any recognition of their role in the compression of information. It seems that these remarks also apply to the use of these techniques in mathematics, as described in Section 7, below.

Other widely-used examples include the way names of things may serve as relatively short codes for relatively complex concepts, and likewise with "content" words in natural language (Appendix **??**).

Appendix C provides some details relating to the frequencies and sizes of repeating patterns, and their codes.

7 How basic techniques for information compression may be seen in the structure and workings of mathematics

This section describes how the basic principles and techniques for the compression of information that were outlined in Section 6 may be seen in the structure and workings of mathematics.

Of course, these examples do not prove that mathematics may be understood as being entirely devoted to the compression of information. But since the techniques to be described are low-level techniques that are part of the foundations of mathematics and widely used in more complex forms of mathematics, it seems

likely that mathematics may indeed be understood in its entirety to be a set of techniques for compressing information and their application.

7.1 ICMUP in mathematics and related fields

Here are some examples where ICMUP may be seen at work in mathematics and related fields:

- In mathematics, the matching and unification of patterns can be seen in the matching and unification of names or identifiers. If, for example, we want to calculate the value of z from these equations: $x = 4$; $y = 5$; $z = x + y$, we need to match the identifier x in the third equation with the identifier x in the first equation, and to unify the two so that the correct value is used for the calculation of z. Likewise for y.

- In a similar way if we wish to invoke or "call" a function such as '`sqrt(x)`' (the square root of x), there must be a match between the name of the function in the call to the function (such as '`sqrt(16)`') and the name of the function in its definition ('`sqrt(x)`'), with unification to assign the value 16 to the variable x.

- The sixth of Peano's axioms for natural numbers—for every natural number n, $S(n)$ is a natural number—provides the basis for a succession of numbers: $S(0)$, $S(S(0))$, $S(S(S(0)))$ ..., itself equivalent to unary numbers in which $1 = /$, $2 = //$, $3 = ///$, and so on. Here, S at one level in the recursive definition is repeatedly matched and unified with S at the next level.

- Emil Post's [1943] "Canonical System", which is recognised as a definition of "computing" that is equivalent to a universal Turing machine, may be seen to work largely via the matching and unification of patterns. Much the same is true of the workings of the transition function in a universal Turing machine.

- It is true that logic gates provide the mechanism for finding an address in computer memory but, at a more abstract level, the process may be seen as one of searching for a match between the address held in the CPU and the corresponding address in computer memory. When a match has been found between the address in the CPU and the corresponding address in memory, there is implicit unification of the two.

- Query-by-example, a popular technique for retrieving information from databases, is essentially a process of finding good matches between a query pattern and patterns in the database, with unification of the best matches.

- A system like Prolog—a computer-based version of logic—may be seen to function largely via the matching and unification of patterns.

7.2 Chunking-with-codes

If a set of statements is repeated in two or more parts of a computer program then it is natural to declare them once as a 'function', 'procedure' or 'sub-routine' within the program and to replace each sequence with a "call" to the function from each part of the program where the sequence occurred. This may be seen as an example of the chunking-with-codes technique for information compression: the function may be regarded as a chunk, while the name of the function is its code or identifier.

Similar things may be done with mathematics, but most of the widely-used functions—such as '`sqrt()`', '`sin()`', or '`cosin()`'—are provided ready-made in environments like Matlab.

Number systems with bases greater than 1, like the binary, octal, decimal and hexadecimal number systems, may all be seen to illustrate the chunking-with-codes technique for compressing information. For example, with the decimal system:

- A unary number like '/////////' may be referred to more briefly as '7'. Here, '/////////' is the chunk and '7' is the code.
- A unary number like '//////////////////' may be split into two parts: '//////////' and '/////////'. Then the first part may be represented by '1' and the second part by '7', giving us the decimal number '17'. The convention is that the right-most digit represents numbers less than 10, and the next digit to the left represents the number of 10s.
- Of course, this 'positional' system can be extended so that digits in the third position from the right represent 100s, digits in the fourth position represent 1000s, and so on.

Here, we can see how the chunking-with-codes technique allows us to eliminate the repetition or redundancy that exists in all unary numbers except '/' so that large numbers, like 2035723, may be expressed in a form that is very much more compact than the equivalent unary number.

7.3 Schema-plus-correction

Most functions in mathematics and computing, like those mentioned above, are not only examples of chunking-with-codes: they are also examples of the schema-plus-correction device for compressing information. This is because they normally

require input via one or more "arguments" or "parameters". For example, the square root function needs a number like 16 for it to work on. Without that number, the function is a very general "schema" for solving square root problems. With a number like 16, which may be regarded as a "correction" to the schema, the function becomes focussed much more narrowly on finding the square root of 16.

7.4 Run-length coding

Run-length coding appears in various forms in mathematics, normally combined with other things. Here are some examples:

- Multiplication (eg, 3×4) is repeated addition.
- Division of a larger number by a smaller one (eg, 12/3) is repeated subtraction. Of course there will be a "remainder" if the larger number is not an exact multiple of the smaller number.
- The power notation (eg, 10^9) is repeated multiplication, which is itself a form of run-length coding.
- A factorial (eg, 25!) is repeated multiplication and subtraction.
- The bounded summation notation (eg, $\sum_{i=1}^{5} \frac{1}{i}$) and the bounded power notation (eg, $\prod_{n=1}^{10} \frac{n}{n-1}$) are shorthands for repeated addition and repeated multiplication, respectively. In both cases, there is normally a change in the value of a variable on each iteration, so these notations may be seen as a combination of run-length coding and schema-plus-correction.
- In matrix multiplication, AB is a shorthand for the repeated operation of multiplying each entry in matrix A with the corresponding entry in matrix B.

All of these examples may be seen as functions with one or more parameters. For example, multiplication may be written $multiply(x, y)$. As functions with parameters, the examples may be seen to illustrate the chunking-with-codes and schema-plus-correction techniques for compressing information (Section 7.3), as well as run-length coding.

8 Well-known equations

The well-known equations that were mentioned earlier may all be interpreted in terms of the first three of our five basic techniques for compressing information, thus:

- Einstein's $E = mc^2$ illustrates run-length coding in its power notation (c^2) and in the multiplication of m with c^2.
- Newton's equation for his second law of motion, $s = (gt^2)/2$, illustrates run-length coding in its power notation (t^2), in the multiplication of g with t^2, and in the division of (gt^2) by 2.
- Pythagoras's equation, $a^2 + b^2 = c^2$, illustrates run-length coding via the power notation in a^2, b^2, and c^2.
- Boyle's law, $PV = k$, illustrates run-length coding in the multiplication of P by V.
- The charged particle equation, $F = q(E+v\times B)$, illustrates run-length coding in the multiplication of v by B and in the multiplication of $(E + v \times B)$ by q.

Since multiplication, the power notation, and division, may each be seen as an example of chunking-with-codes and schema-plus-correction (Sections 7.2 and 7.3), as well as run-length coding (Section 7.4), the same can be said about the appearance of those notations in each of the examples above.

9 The apparent paradox of creating redundancy via information compression

The idea that mathematics or computing is largely, perhaps entirely, about compression of information may seem to conflict with the undoubted fact that, with some simple mathematics or a simple computer program, it is possible to create data containing large amounts of repetition or redundancy. An example is shown in Figure 1.[4]

This recursive function takes an integer like 5 as its parameter and prints out the same number of copies of '`hello, world`', like this: '`hello, world; hello, world; hello, world; hello, world; hello, world;`'.

[4] A tribute to Brian Kernighan and Dennis Richie's [1988] introduction to the C programming language.

```
void create\_redundancy(int x)
{
    if (x <= 0) return ;
    printf("hello, world; ") ;
    return create\_redundancy(x - 1) ;
}.
```

Figure 1: A simple recursive function showing how, via computing, it is possible to create repeated (redundant) copies of "hello, world".

With this and similar examples, there seems to be a paradox in that a system that supposedly works via the compression of information is able to produce an output that clearly contains significant amounts of redundancy. The suggestion here is that the paradox is more apparent than real and may be resolved as follows:

- Like any other function with one or more parameters, the example function, and the '`printf()`' function within it, illustrate the principles of chunking-with-codes and schema-plus-correction, as described in Sections 7.2 and 7.3.
- In its workings, the function requires the matching and unification of identifiers like '`create_redundancy`' (in the name of the function and in the third line of the function) and '`printf`' (in the second line of the function and in the operating system that does things like 'printing' messages on the computer screen).
- The entire function may be seen to be a compressed version of the output sequence and any similar sequence however long it may be, in much the same way that $s = (gt^2)/2$ is a compressed version of Table 1 and any similar table, however large it may be (Section 5).

There is discussion of a related issue—how the SP system may achieve "decompression by compression"—in Wolff [2006, Section 3.8] and Wolff, Section 4.5.

10 Redundancy is often useful in the storage and processing of information

There is no doubt that informational redundancy—repetition of information—is often useful. For example:

- With any kind of database, it is normal practice to maintain one or more backup copies as a safeguard against catastrophic loss of the data.

- With information on the internet, it is common practice to maintain two or more 'mirror' copies in different places to minimise transmission times and to reduce the chance of overload at any one site.

- The redundancy in natural language can be a very useful aid to comprehension of speech in noisy conditions.

These kinds of uses of redundancy may seem to conflict with the idea that information compression—which means reducing redundancy—is fundamental in mathematics, computing and cognition. However, the two things may be independent, or the usefulness of redundancy may actually be understood in terms of the SP theory itself.

An example of how the two things may be independent is the above-mentioned use of backup copies of databases: "... it is entirely possible for a database to be designed to minimise internal redundancies and, at the same time, for redundancies to be used in backup copies or mirror copies of the database ... Paradoxical as it may sound, knowledge can be compressed and redundant at the same time." [Wolff, 2006, Section 2.3.7].

How the usefulness of redundancy may be understood in terms of the SP theory is discussed in [Wolff, Sections 8 and 9], Wolff [2007], [Wolff, 2006, Section 6.2]. In brief:

- In the retrieval of information from a database or other body of knowledge, there needs to be some redundancy between the search pattern and each matching pattern in the knowledge base.

- And redundancy provides the key to how, in applications such as pattern recognition and the parsing of natural language, the SP system may achieve good results despite errors of omission, commission or substitution and thus, in effect, suggest interpolations for errors of omission and corrections for errors of commission or substitution.

11 Mathematical "beauty" and information compression

In a paper with the title "Driven by compression progress: a simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes" [Schmidhuber, 2009], and in several earlier papers, Schmidhuber describes how mathematical "beauty", amongst other things, may be understood in terms of the compression of information. His analysis, which is largely in terms of algorithmic information theory

and related concepts [Li and Vitányi, 2014], is somewhat different from what has been described in Sections 5 to 9, above, which attempts to reach down to relatively "primitive" concepts like the discovery of matches between patterns and the merging or "unification" of patterns that are the same.

12 Probabilities

If it is accepted that information compression is central in the structure and workings of mathematics, then in view of the intimate connection between information compression and concepts of prediction and probability (Section 3.2 and Appendix D), there are reasons to think that mathematics is fundamentally probabilistic. Although this seems to conflict with the apparent certainty of equations like $2+2 = 4$, a probabilistic foundation for mathematics is consistent with the discovery of randomness in number theory:

> "I have recently been able to take a further step along the path laid out by Gödel and Turing. By translating a particular computer program into an algebraic equation of a type that was familiar even to the ancient Greeks, I have shown that there is randomness in the branch of pure mathematics known as number theory. My work indicates that—to borrow Einstein's metaphor—God sometimes plays dice with whole numbers." [Chaitin, 1988, p. 80].

As indicated in this quotation, randomness in number theory is closely related to Gödel's incompleteness theorems. These are themselves closely related to the phenomenon of recursion, a feature of many formal systems, several of Escher's pictures, and much of Bach's music, as described in some detail by Douglas Hofstadter in *Gödel, Escher, Bach: An Eternal Golden Braid* [Hofstadter, 1980].

13 Mathematics as compression of information and the philosophy of mathematics

Amongst the several "isms" in the philosophy of mathematics—foundationism, logicism, intuitionism, formalism, Platonism, neo-Fregeanism, and more—the three which are perhaps most closely related to MAIC are *psychologism* (mathematical concepts derive from human psychology), *embodied mind theories* (mathematical thought is a natural outgrowth of human cognition), and *intuitionism* (mathematics is a creation of the human mind).

Appendix A outlines some of the evidence in support of the view that much of human learning, perception and cognition may be understood as compression of information. This is broadly consistent with the three schools of thought mentioned above.

Probably the most distinctive feature of MAIC is that it does not, to my knowledge, feature in any of psychologism, embodied mind theories, intuitionism, or any other school of thought in the philosophy of mathematics. Also, MAIC may be seen to apply not only to human thinking but also to varied kinds of artificial device for the processing of information.

14 An apparent solution to the mystery of why mathematics is so effective in science

In view of evidence that: 1) science is fundamentally a search for compression in the world (Section 3); and evidence that 2) mathematics may be seen to be largely a set of techniques for compressing information and their application (Section 4); and bearing in mind 3) the afore-mentioned intimate relation between information compression and concepts of prediction and probability (Appendix D); it seems reasonable to conclude that those three things may explain why mathematics is so effective as a means of representing scientific knowledge and in the making of scientific inferences.

There is relevant discussion in Appendices B and D.

15 The anthropic principle

An objection to the arguments in Section 14 is that, while $E = mc^2$ is undoubtedly a compressed representation of the data that it describes, that observation does not explain why nature can be so compressible.

A possible answer, via the anthropic principle, is that the world must be compressible because otherwise everything in it, including ourselves, would be a soup of randomness—meaning that we would not be here to observe things.

Of course, some aspects of the world are more compressible than others. In social sciences, for example, it has proved difficult to find equations that are so elegantly simple and powerful as $E = mc^2$.

16 Conclusion

This paper notes first that the effectiveness of mathematics in science appears to some writers to be "mysterious" or "unreasonable". Then reasons are given for thinking that science is fundamentally a search for compression of empirical data. At more length, several reasons are given for believing that mathematics is, fundamentally, a set of techniques for compressing information—including the matching and unification of patterns, chunking-with-codes, schema-plus-correction, and run-length coding—and their application. From there, it is argued that mathematics has proved to be effective in science because it provides a means of achieving the compression of information which lies at the heart of science.

Information compression may be seen to be important in both science and mathematics, not only as a means of representing knowledge succinctly, but as a basis for scientific and mathematical inferences—because of the intimate relation that is known to exist between information compression and concepts of prediction and probability.

That mathematics may be seen to be a set of techniques for compressing information and their application, is in keeping with the view, supported by evidence, that much of human learning, perception, and cognition may understood as the compression of information.

Acknowledgements

I'm grateful to Roger Penrose and John Barrow for helpful comments on this paper. For helpful comments on drafts of an earlier related paper, I'm grateful to Robert Thomas, Michele Friend, and Alex Paseau. I'm also grateful for discussion from time to time of some of the ideas in this paper with Tim Porter and Chris Wensley.

A Intelligence as compression of information

As noted in the Introduction, much of the thinking in this paper derives from the *SP theory of intelligence* and its realisation in the *SP computer model*. This theory, which is described quite fully in Wolff [2006] and more briefly in Wolff, aims to simplify and integrate observations and concepts across artificial intelligence, mainstream computing, mathematics, and human learning, perception, and cognition.

The SP system is founded on a considerable body of evidence pointing to the significance of information compression in the workings of brains and nervous systems, in language learning, in concepts of prediction and probability, and in

solving problems in science and mathematics. Some of this evidence is summarised in Wolff [1993]. That paper provides the basis for [Wolff, 2006, Chapter 2]. A revised and updated account may be found in Wolff [2017].

Central to the workings of the SP system is information compression via the matching and unification of patterns, and more specifically, information compression via the powerful concepts of *multiple alignment*, a concept that has been borrowed and adapted from bioinformatics.

The success of the SP system in modelling several aspects of human learning, perception, and cognition [Wolff, 2016], provides additional evidence for the significance of information compression for an understanding of *intelligence*, broadly construed.

B Generalisation in science

Science is not merely about describing things in an economical manner, it is about making predictions or inferences that go beyond what has actually been observed. As described in Appendix D, there is an intimate connection between information compression and concepts of prediction and probability.

This appendix summarises some ideas discussed elsewhere (Wolff [2006, Section 2.2.12], Wolff, Section 5.3) about how we can or should generalise our concepts without over-generalisation (sometimes called under-fitting) or under-generalisation (sometimes called over-fitting).

This issue is important in understanding how a child learns his or her first language or languages. The learning of a given language, **L**, is based on a finite sample of **L** that has been heard, normally quite large. This is represented by the smallest envelope in Figure 2, marked as "A sample of utterances".[5] From that finite sample, we learn to understand and to produce a range of possible utterances that is much larger than the finite sample we have heard. This is represented by the envelope marked "All utterances in language L". But although that range of utterances is large, it is smaller than the range of all possible utterances represented by the envelope marked "All possible utterances". Notice that the smallest envelope—the basis for learning—is partly inside the envelope for "All utterances in language L" and partly outside it: children learn partly from good examples

[5]The weight of evidence is overwhelmingly against the nativist, Chomskian view that children are born with substantial knowledge of the structure of language. Some of the evidence is described in Wolff [1988, pp. 208–209 and pp. 210-211]. Perhaps the strongest argument, not made in that publication, is that, to explain why a newborn baby can learn any natural language, the nativist view depends on the existence of some kind of *universal grammar* that describes the structure of every one of the thousands of natural languages and is in every infant's head at the time of his or her birth. Despite decades of research, no such universal grammar has been found.

of **L** and partly from corrupted examples of **L**, which are marked in the figure as "dirty data".

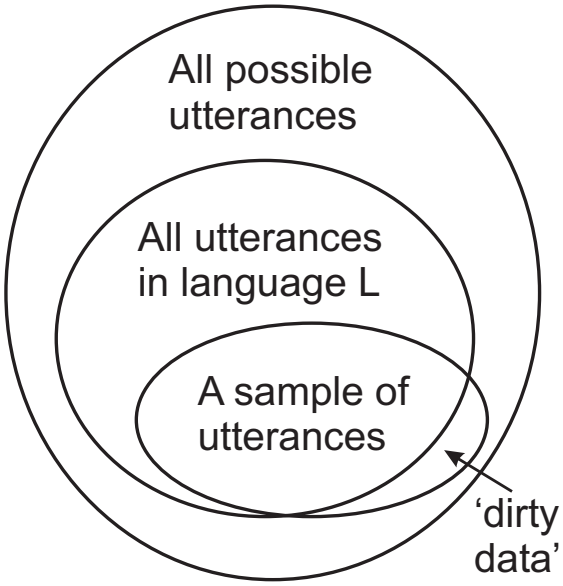

Figure 2: Categories of utterances involved in the learning of a first language, **L**. In ascending order size, they are: the finite sample of utterances from which a child learns; the (infinite) set of utterances in **L**; and the (infinite) set of all possible utterances. Adapted from Figure 7.1 in Wolff [1988], with permission.

In summary, learning **L** means generalising beyond the finite sample of utterances that we have heard but without either over-generalisation or under-generalisation, and it means somehow correcting for dirty data. Although young children say things like "gooses" and "sheeps"—apparently overgeneralising from what they have heard—they grow out of these errors. The weight of evidence is that children can learn a first language without the need for explicit correction of errors by adults or older children.[6]

There is evidence that learning of **L** can be achieved, without over-generalisation or under-generalisation, correcting for dirty data, and without the need for explicit correction of errors. Here's how:

1. Start with a finite sample of **L** which we may call **I**. The sample may contain dirty data as described above.

[6]Christy Brown was a cerebral-palsied child who not only lacked any ability to speak but whose bodily handicap was so severe that for much of his childhood he was unable to demonstrate that he had normal comprehension of speech and non-verbal forms of communication [Brown, 2014]. Hence, his learning of language must have been achieved without the possibility that anyone might correct errors in his spoken language.

2. Compress **I** using something like the SP system, designed to achieve high levels of lossless information compression.[7]

3. The result of compressing **I**, which we may call **IC**, may be divided into two parts: a 'grammar' **G**, and an 'encoding' of **I** in terms of **G**, which we may call **E**.

4. Of **G** and **E**, the most interesting seems normally to be **G**, which may be regarded as a 'theory' of **I**. It appears that, normally, **G** represents a distillation of the 'essence' of **I**, weeding out any dirty data in **I**, and generalising from **I** without overgeneralising. **E** is a description of **I** in terms of the theory, **G**.

This solution for language learning appears to be general and applicable to the learning of any kind of kind of knowledge. It seems likely that, for example, it will provide a solution to the problem of how, via unsupervised learning, a concept like 'horse' may be learned without under-generalisation (meaning that, for example, the system would only recognise horses that are very similar to, or identical with, the examples of horses in **I**), and without over-generalisation (meaning that, for example, the system would regard cows, sheep, or dogs, as horses).

It appears that this solution is altogether simpler and more comprehensive than several alternatives, as discussed in Wolff [2016, Section V-H].

Without such generalisation, any learning system would be severely handicapped: only able to recognise or understand things that were exactly the same as it had seen before.

C Frequency of occurrence, sizes of patterns, and ICMUP

A point to notice about ICMUP is that, to achieve lossless compression of information, it is necessary to use some kind of "code" to mark the positions of redundant copies of any pattern that have been unified, as outlined under the heading "Chunking-with-codes" in Section 6.1. And to ensure that there is an overall compression of a given body of information, **I**, it is necessary to ensure that:

[7] Ordinary compression algorithms, like the popular 'ZIP' algorithms, are not really suitable because they are designed to work fast with low-powered computers and may thus miss relatively large amounts of redundancy.

- The given pattern must repeat more often in **I** than we would expect by chance *for patterns of that size*. In general, large patterns yield more compression than small ones, and the minimum frequency needed to achieve information compression is smaller for large patterns than it is for small patterns.
- The code should not be too large. Normally, its size should be at or near the theoretical minimum needed to ensure an overall compression of **I**.

These points relate to the close connection between information compression and concepts of prediction and probability, discussed in Appendix D.

D Information compression and concepts of prediction and probability

It has been recognised for some time that there is an intimate connection between information compression and concepts of prediction and probability, as described in Ray Solomonoff's Algorithmic Probability Theory [Solomonoff, 1964, 1997], and in the closely-related Kolmogorov Complexity Theory [Li and Vitányi, 2014]. Information compression and concepts of prediction and probability may be seen as two sides of the same coin.

The close connection between those two things makes sense in terms of ICMUP (Section 6):

- A pattern that repeats is one that, via inductive reasoning, we naturally regard as a guide to what may happen in the future (more in Appendix D.2, below).
- A pattern that repeats is one that, via unification, is likely to yield compression of information.
- A partial match between one pattern and another can be the basis for predicting the occurrence of the unmatched parts, a form of inference that is sometimes called *prediction by partial matching*.[8]

D.1 Frequencies of occurrence, sizes of patterns, and probabilities

A point of interest is that, in the same way that information compression depends partly on the frequency of occurrence of a pattern that repeats, and also on its

[8] See "Prediction by partial matching", *Wikipedia*, bit.ly/1BUtAYo, retrieved 2017-03-01.

size (Appendix C), the probabilities of inferences that may be drawn from any repeating pattern depend partly on the frequency of occurrence of the given pattern and partly on its size. How relevant calculations are made in the SP system is described in Wolff [2006, Sections and 3.7 7.2].

More specifically, a repeating pattern of size **S** can only yield inferences with probabilities greater than chance if its frequency of occurrence within a given body of information, **I**, is greater than the 'threshold' frequency of occurrence that would be expected by chance *for a pattern of size* **S**; and that threshold frequency is greater for small patterns than it is for large patterns.

Consider, for example, what inferences one might make from the occurrence, in an English text, of the neighbouring letters, '`th`'. Given only those two letters, one may guess that they may be part of several different words such as '`the`', '`this`', '`that`', '`those`', and so on, each one with a probability that is substantially less than 1. But although, notwithstanding its fame, the pattern of words, '`Let me not to the marriage of true minds`', is much rarer in English than the pattern '`th`', we infer with near certainty that it will be followed by the words '`Admit impediments`'.

With regard to frequencies and sizes of patterns in the calculation of probabilities:

- There is a sharp contrast between the SP system, which takes account of both the frequencies and the sizes of patterns in calculating probabilities, and frequentist approaches to statistics which emphasise the frequencies of occurrence of entities, without taking account of their sizes.

- "Hebbian" learning, first proposed by neuroscientist Donald Hebb [1949, p. 62], with a central role in most versions of "deep learning" [Schmidhuber, 2015], is focussed entirely on the frequency with which one neuron fires another, without any role for the sizes of neural structures involved in learning, perception and cognition.

D.2 Inductive reasoning

With regard to inductive reasoning mentioned in the first bullet point in Appendix D:

> "We can, of course, ... ask, as philosophers have done for many years: 'What is the rational basis for inductive reasoning?' Why do most people have this strong intuition that because the sun has always risen every morning it will do it again tomorrow, or because every paving stone in a path has held our weight so far, the next one will too? None of these conclusions can be proved logically.

"It is no good arguing that inductive reasoning is rational because it has always worked in the past. This argument eats its own tail. Here is an argument why inductive reasoning is rational which does not depend on the principle which it is trying to justify:

> "If we assume that the world, in the future, will contain redundancy in the form of recurring patterns of events, then brains and computers which store information and make inductive inferences will be useful in enabling us to anticipate events. If it turns out that the world, in the future, does indeed contain redundancy then our investment in the means of storing and processing information will pay off. If it turns out that the world, in the future, does not contain redundancy then we are dead anyway—reduced to a pulp of total chaos!

"This kind of reasoning made fortunes for speculators after World War II: it was rational to buy up London bomb sites during the war because, if the war were won, they would become valuable. If the war were to be lost, the money saved by not making the investment would, in an uncomfortable and uncertain future, probably not be much use anyway." [Wolff, 1991, pp. 28–29].

References

E. C. Banks. The philosophical roots of Ernst Mach's economy of thought. *Synthese*, 139:25–53, 2004.

J. D. Barrow. *Pi in the Sky*. Penguin Books, Harmondsworth, 1992.

E. T. Bell. *The Handmaiden of the Sciences*. The Williams & Wilkins Company, Baltimore, 1937.

C. Brown. *My Left Foot*. Vintage Digital, London, Kindle edition, 2014. First published in 1954.

G. J. Chaitin. Randomness in arithmetic. *Scientific American*, 259(1):80–85, 1988.

D. O. Hebb. *The Organization of Behaviour*. John Wiley & Sons, New York, 1949.

D. Hofstadter. *Gödel, Escher, Bach: an Eternal Golden Braid*. Penguin Books, Harmondsworth, 1980.

W. Isaacson. *Einstein: His Life and Universe*. Pocket Books, London, Kindle edition, 2007.

B. W. Kernighan and D. M. Ritchie. *The C Programming Language*. Prentice Hall, Englewood Cliffs, NJ, second edition, 1988.

M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer, New York, 3rd edition, 2014.

I. Newton. *The Mathematical Principles of Natural Philosophy*. The Perfect Library, Kindle edition, 2014. First published 1687. Illustrated and bundled with *Life of Sir Isaac Newton*.

K. Pearson. *The Grammar of Science*. Walter Scott, London, 1892. Republished by Dover Publications, 2004, ISBN 0-486-49581-7. Internet archive: bit.ly/1g2gNfk.

R. Penrose. *The Emperor's New Mind*. Oxford University Press, Oxford, 1989.

E. L. Post. Formal reductions of the general combinatorial decision problem. *American Journal of Mathematics*, 65:197–268, 1943.

C. Rovelli. *Reality Is Not What It Seems: The Journey to Quantum Gravity*. Penguin Books, London, kindle edition, 2016.

J. Schmidhuber. Driven by compression progress: a simple principle explains essential aspects of subjective beauty, novelty, surprise, interestingness, attention, curiosity, creativity, art, science, music, jokes. In G. Pezzulo, M. V. Butz, O. Sigaud, and G. Baldassarre, editors, *Anticipatory Behavior in Adaptive Learning Systems, from Sensorimotor to Higher-level Cognitive Capabilities*, Lecture Notes in Artificial Intelligence. Springer, Berlin, 2009.

J. Schmidhuber. Deep learning in neural networks: an overview. *Neural Networks*, 61:85–117, 2015. doi: 10.1016/j.neunet.2014.09.003.

R. J. Solomonoff. A formal theory of inductive inference. Parts I and II. *Information and Control*, 7:1–22 and 224–254, 1964.

R. J. Solomonoff. The discovery of algorithmic probability. *Journal of Computer and System Sciences*, 55(1):73–88, 1997.

E. Wigner. The unreasonable effectiveness of mathematics in the natural sciences. *Communications in Pure and Applied Mathematics*, 13:1–14, 1960.

J. G. Wolff. The SP theory of intelligence: an overview. *Information*, 4(3):283–341. doi: 10.3390/info4030283.

J. G. Wolff. Learning syntax and meanings through optimization and distributional analysis. In Y. Levy, I. M. Schlesinger, and M. D. S. Braine, editors, *Categories and Processes in Language Acquisition*, pages 179–215. Lawrence Erlbaum, Hillsdale, NJ, 1988. bit.ly/ZIGjyc.

J. G. Wolff. *Towards a Theory of Cognition and Computing.* Ellis Horwood, Chichester, 1991.

J. G. Wolff. Computing, cognition and information compression. *AI Communications*, 6(2):107–127, 1993. bit.ly/XL359b.

J. G. Wolff. *Unifying Computing and Cognition: the SP Theory and Its Applications.* CognitionResearch.org, Menai Bridge, 2006. ISBNs: 0-9550726-0-3 (ebook edition), 0-9550726-1-1 (print edition). Distributors, including Amazon.com, are detailed on bit.ly/WmB1rs.

J. G. Wolff. Towards an intelligent database system founded on the SP theory of computing and cognition. *Data & Knowledge Engineering*, 60:596–624, 2007. doi: 10.1016/j.datak.2006.04.003. bit.ly/1CUldR6.

J. G. Wolff. The SP theory of intelligence: its distinctive features and advantages. *IEEE Access*, 4:216–246, 2016. doi: 10.1109/ACCESS.2015.2513822. bit.ly/2qgq5QF.

J. G. Wolff. Information compression in brains and nervous systems. Technical report, CognitionResearch.org, 2017. In preparation.

G. K. Zipf. *Human Behaviour and the Principle of Least Effort.* Hafner, New York, 1949. Republished by Martino Publishing, Mansfield Centre, CT, 2012.