Transreal arithmetic as a consistent basis for paraconsistent logics

Conference or Workshop Item

Accepted Version

Transreal Arithmetic as a Consistent Basis For Paraconsistent Logics

James A.D.W. Anderson and Walter Gomide

Author Final Version: July 25, 2014

J.A.D.W. Anderson is with the School of Systems Engineering, Reading University, England, RG6 6AY e-mail: j.anderson@reading.ac.uk

Walter Gomide is with the Federal University of Mato Grosso (UFMT), Brazil e-mail: waltergomide@yahoo.com

Abstract

Paraconsistent logics are non-classical logics which allow non-trivial and consistent reasoning about inconsistent axioms. They have been proposed as a formal basis for handling inconsistent data, as commonly arise in human enterprises, and as methods for fuzzy reasoning, with applications in Artificial Intelligence and the control of complex systems.

Formalisations of paraconsistent logics usually require heroic mathematical efforts to provide a consistent axiomatisation of an inconsistent system. Here we use transreal arithmetic, which is known to be consistent, to arithmetise a paraconsistent logic. This is theoretically simple and should lead to efficient computer implementations.

We introduce the metalogical principle of monotonicity which is a very simple way of making logics paraconsistent.

Our logic has dialetheaic truth values which are both False and True. It allows contradictory propositions, allows variable contradictions, but blocks literal contradictions. Thus literal reasoning, in this logic, forms an on-the-fly, syntactic partition of the propositions into internally consistent sets. We show how the set of all paraconsistent, possible worlds can be represented in a transreal space. During the development of our logic we discuss how other paraconsistent logics could be arithmetised in transreal arithmetic.

Keywords: transreal arithmetic, paraconsistent logic.

1 Introduction

Paraconsistent logics were explicitly introduced in the second half of the twentieth century as non-classical logics that can reason about inconsistent axioms [26][13]. In a classical logic, inconsistent axioms *explode*, allowing any theorem to be proved in a trivial way [11]. By contrast paraconsistent logics do not explode, they allow only limited conclusions to be drawn from inconsistent axioms. Some paraconsistent logics admit *dialetheias*, that is propositions which are both False and True [24], and some admit Gap values with no component of falsity or truthfulness [29]. Gap values are usually treated absorptively so that any logical combination with a Gap produces a Gap as result. This behaviour is consistent with one reading [20] of Frege's principle of compositionallity so that a compound proposition lacks reference if any component of it lacks reference. It should be added that paraconsistent logics are also capable of classical reasoning so they provide a robust generalisation of classical logic. This makes them interesting both from a theoretical and a practical perspective.

It has been proposed that paraconsistent logics are suitable for implementing fuzzy logical systems, including the control of complex systems [34][10], for practical reasoning about inconsistent data, such as the data typically provided by humans, for use in Artificial Intelligence programs, for human use in developing scientific theories that contain contradictory elements, and more [26].

Paraconsistent logics are usually formalised in terms of advanced mathematics. Shramko [29] reviews some logics using small, discrete, ordered sets of truth values; these are mathematically simple, though one of the logics uses two orderings. Logics with more than two orderings are discussed in [30]. Some paraconsistent logics partition propositions into a hierarchy of internally consistent partitions.

Here we take the simpler approach of expressing a paraconsistent logic arithmetically. We use transreal arithmetic, which is a generalisation of real arithmetic. Transreal arithmetic was originally developed [2][3] from a subset of the algorithms used in the arithmetic of fractions. It has been axiomatised and a machine proof of consistency has been given [8]. Two human proofs of consistency are known but neither has been published to date. The algorithms of transreal arithmetic are explained, particularly clearly, in a recent treatment, in which transcomplex numbers are also introduced [5].

We develop a paraconsistent logic by expressing the Sheffer Stroke [9] as an operation in transreal arithmetic. The Sheffer Stroke can be used to develop all classical, truth functional logics [9] (See entry "Sheffer Stroke"), [28] (p. 29) so our paraconsistent logic is just one example from an entire class of paraconsistent generalisations of classical logic. During the development of our logic we explain the motivation for each of the design decisions and consider alternatives so as to assist readers in developing their own paraconsistent logics that use transreal arithmetic as a consistent basis.

The Sheffer Stroke is better known, in Electronic Engineering, as the Not-And operation, NAND [28]. It can be used to develop all of the logic circuits in digital computers. By generalising the NAND gate to paraconsistent form, we create the possibility of fabricating paraconsistent processors in hardware. We briefly consider features of high-level, computer languages that implement paraconsistent logics. This may be of interest to practitioners of Artificial Intelligence and Cybernetics (control theory).

2 Paraconsistent Logic

2.1 Truth Values

The transreal numbers are just the real numbers augmented with three non-finite numbers: negative infinity ($-\infty$), positive infinity (∞) and nullity (Φ). Nullity is absorptive over the elementary arithmetical operations so that when it is involved in a sum, difference, product or quotient, the result is nullity. However nullity is not universally absorptive, it may be an element of arbitrary mappings. Nullity is the only unordered number in transreal arithmetic [8][7].

Nullity's absorptive properties make it a good candidate for a Gap value that has no degree of falsity or truthfulness.

The utility of a Gap value can be illustrated with the well known fairy story of *Goldilocks and the Three Bears.* Goldilocks is interested in predicates concerning porridge. She wants to know the truth values of too_cold(porridge), too_hot(porridge), just_right(porridge). But suppose Goldilocks' reasoning system is presented with the logical argument porridge, devoid of a predicate. What is she to do? The argument porridge is a signifier for boiled oats that lie in a bowl in front of her. Neither the signifier nor the actual boiled oats are in the class of things that can be assigned degrees of falsity or truthfulness. Reflecting on her difficulty, Goldilocks may decide to assign a Gap value, that is a value with no degree of falsity or truthfulness, to the signifier porridge. Going further she may decide to apply Gap values to all badly formed formulas of logic and to everything that cannot be assigned a degree of falsity or truthfulness, such as actual boiled oats. This makes Goldilocks' reasoning system total. She can assign a logical value to every possible sentence, including meta sentences that consider the properties of actual objects, such as boiled oats, being presented directly to her reasoning system. Notice that we must say "sentence" here, not "predicate," if we are to admit badly formed formulae. We hope we have done enough to convey the utility of a Gap value to the reader.

We define that negative infinity is classical False and positive infinity is classical True. This has the merit that we have now used up all of the non-finite, transreal numbers, leaving all of the real numbers to convey dialetheaic degrees of falsity and truthfulness.

In the next subsection we use arithmetical negation (unary subtraction) to model logical negation, thereby exploiting the following idempotence of transreal arithmetic: $-(\infty) = -\infty, -(-\infty) = \infty$. Many, perhaps most, paraconsistent logics will use some kind of idempotent negation but, perhaps, one that cycles through many truth values, not just two. But not all logics have an idempotent negation. For example many computer languages use zero for False and any non-zero value for True. This is exploited in cases, such as memory management, where an applications programmer instructs the assignment of memory and is returned a single value by the operating system. If the returned value is non-zero, it is True that the operating system has assigned memory and the returned value is the base address of that memory but if the returned value is zero then it is False that the operating system has assigned memory. In such languages the negation of any True value is the unique False value but the negation of the unique False value is exactly one, fixed one, of the True values. In this case negation is not idempotent on the truth values but it is idempotent on the classes from which the specific truth values are drawn. One might even want continuous forms of negation implemented, say, as rotations in the complex plane, or one might want continuous blends of negation with other operators [4]. Yet other kinds of negation might be wanted.

If we were to model logical negation with the arithmetical reciprocal then we might chose to model False with zero and True with infinity so that negation is supplied by the transreal idempotence $1/0 = \infty, 1/\infty = 0$. This approach is taken by Gomide [18]; while logically sufficient, the operators are not total functions of the transreal numbers, which complicates the application of advanced mathematics to such a system. The computation of reciprocals also has more numerical error or higher computational cost than the arithmetisation used here. Of course logical negation can be modelled in many other arithmetical ways, including by arbitrary mappings.

If a fuzzy or statistical system is wanted then we work with probabilities, in which case transreal numbers outside the range from zero to one, inclusive, arise only in the underlying frequencies. Recall [17][15] that a statistical frequency, $f = o/e$, is the ratio of the number of target outcomes, o, to the number of experiments, e. This admits all non-negative, rational frequencies. The non-negative, irrational frequencies are admitted via probability density functions. The non-negative, non-finite, transreal numbers are also valid frequencies. Nullity, $\Phi = 0/0$, is the frequency where no target outcomes have occurred in no experiments, say where a coin is held on its edge, between two leaves of a table, before it is tossed, in an experiment to determine whether heads lie face-up on the coin when it comes to rest. Infinity, $\infty = k/0 = 1/0$, for all positive k, is the frequency where a positive number of outcomes, say one, occurs in no experiments, say where the frequency of heads is wanted in a coin tossing experiment, where the coin lies heads-up on a table before it is tossed for the first time.

These examples might strike the reader as artificial but they are needed to make statistics total. The reader might be more satisfied by an example that commonly occurs in statistical packages: what is the arithmetical mean of a list of no numbers? Suppose the number of elements, in the list, is accumulated in the variable n and the sum is accumulated in the variable s then the mean is computed as s/n. We program defensively by setting the accumulators to an initial value of zero: $n = s = 0$. When the program handles an empty list, n and s are not incremented, which correctly records that there are no, that is zero, elements in the list and that they have no, that is zero, sum. The program then computes the mean as $n/s = 0/0 = \Phi$. No special handling of the empty list is required, no matter what complexity of statistical computation follows. Thus transreal arithmetic simplifies computer code, removes all arithmetical exceptions from syntactically correct, transarithmetical sentences and provides a consistent basis for fuzzy and statistical, paraconsistent logics.

Returning now to our paraconsistent logic, we define that the real numbers encode degrees of both falsity and truthfulness. The negative real numbers are more False than True, the positive, real numbers are more True than False, zero is equally False and True. We relate the degree of falsity and truthfulness monotonically to the number modelling the truth value so that negative infinity is entirely False, that is classically False, and positive infinity is entirely True, that is classically True. The reader might want to impose specific monotonic functions so as to obtain particular metrics. In this regard the transreal arctangent function [7] might be helpful because it maps a linear, transreal angle onto nullity and the whole range of transreal numbers from negative infinity to positive infinity. Nullity is the unique Gap value. Thus all transreal numbers are used in our paraconsistent logic, making it both total and consistent. (The reader, being educated in logical terminology, will not confuse *total* and consistent with *complete* and consistent.)

2.2 Metalogical Principle of Monotonicity

We have a metalogical intuition that a conclusion can depart no more from being equally False and True than the most divergent of its premises. No matter how an inference is constructed, this is sufficient to cut off all those conclusions that are more divergent so the inference is generally non-explosive.

In our model the absolute value of the conclusion can be no greater than the greatest absolute value of its premises. We operate on absolute values, not signed values, to allow the possibility of a conclusion being the negation of some one or more of its premises. Indeed we are forced into this strategy by implementing our paraconsistent logic in terms of the Sheffer Stroke which involves a negation.

Enforcing metalogical monotonicity on all logical operators is a very simple way to make a logic paraconsistent.

2.3 Sheffer Stroke

It is known that the truth functional (Boolean) operators for logical negation (not, $\neg$), logical conjunction (and, &), and logical disjunction (or, $\vee$) are functionally complete [9] (See entry "Sheffer Stroke"), [28] (p. 29) so that any truth functional operators can be derived from these three. In fact it is known that the sets $\{\neg, \&\}$ and $\{\neg, \vee\}$ are each functionally complete but it serves our purpose better to consider the wider set of operators $\{\neg, \&, \vee\}$.

We begin by introducing the transreal minimum and maximum functions, which we use to define paraconsistent versions of the classical negation, conjunction and disjunction operators. We use negative infinity ($-\infty$) to model classical False (F) and positive infinity (∞) to model classical True (T). We use nullity (Φ) to model the logical Gap value (G). Note that only the real numbers model dialetheaic truth values. The three non-finite numbers each model a single truth value: $-\infty$ models classical False, ∞ models classical True, Φ models Gap. We then prove that the paraconsistent operators contain the classical ones. With a little extra work we prove that the paraconsistent operators are well defined for all transreal arguments when we assume that the finite, truth values are arranged monotonically with the real numbers that model them. We then define a paraconsistent version of the Sheffer Stroke ($|$). There are three, well known, identities that relate the classical Sheffer Stroke to classical negation, conjunction and disjunction. We show that these identities hold when we substitute the paraconsistent Sheffer Stroke and the paraconsistent negation, conjunction and disjunction. Thus we prove that the paraconsistent operators are defined everywhere and are consistent with their classical counterparts.

We begin by defining the binary, transreal, minimum and maximum functions so that the minimum of two transreal numbers is the least, ordered one of them or else is nullity. Similarly the maximum of two transreal numbers is the greatest, ordered one of them or else is nullity. These definitions rely on the three transreal relations less-than, equal-to, greater-than as axiomatised in [8] and explicated in [6]. It is sufficient for the reader to know that: nullity is the uniquely unordered, transreal number so it is the only transreal number that compares not-less-than, not-equal-to and not-greater than any other distinct number; negative infinity is the least, ordered, transreal number; positive infinity is the greatest, ordered, transreal number.

Definition 1. *Transreal minimum,*

$$\min(a,b) = \begin{cases} a & : \quad a < b \\ a & : \quad a = b \\ a & : \quad b = \Phi \\ b & : \quad b < a \\ b & : \quad a = \Phi \end{cases} .$$

Definition 2. *Transreal maximum,*

$$\max(a,b) = \begin{cases} a & : \quad a > b \\ a & : \quad a = b \\ a & : \quad b = \Phi \\ b & : \quad b > a \\ b & : \quad a = \Phi \end{cases} .$$

The minimum and maximum functions, just defined, treat nullity non-absorptively but we chose to treat the logical Gap value absorptively.

Definition 3. *Paraconsistent conjunction,*

$$a \;\&\; b = \begin{cases} \Phi & : \quad a = \Phi \text{ or } b = \Phi \\ \min(a,b) & : \quad \text{otherwise} \end{cases} .$$

Definition 4. *Paraconsistent disjunction,*

$$a \;\vee\; b = \begin{cases} \Phi & : \quad a = \Phi \text{ or } b = \Phi \\ \max(a,b) & : \quad \text{otherwise} \end{cases} .$$

We now define the paraconsistent, logical negation as transarithmetical negation.

Definition 5. *Paraconsistent negation,* $\neg a = -a$.

Transreal arithmetic has $-0 = 0$, $-\Phi = \Phi$ and in all other cases, the negation is distinct so that $-a \neq a$.

The Sheffer Stroke $(|)$ may be defined as an infix operator but we follow the more modern practice of taking it as a post-fix operator so that no bracketing is needed. This leads to shorter and clearer formulae.

Definition 6. *Paraconsistent Sheffer Stroke,* $ab| = \neg(a \;\&\; b)$*, with all symbols read paraconsistently.*

We now prove that the paraconsistent negation, conjunction and disjunction contain their classical counterparts and that the paraconsistent operators are well defined for all transreal arguments.

Theorem 7. *Paraconsistent negation contains classical negation.*

Proof. Classical negation has $\neg\mathrm{F} = \mathrm{T}$ and $\neg\mathrm{T} = \mathrm{F}$. Equivalently paraconsistent negation has $\neg(-\infty) = -(-\infty) = \infty$ and $\neg\infty = -\infty$. □

Theorem 8. *Paraconsistent conjunction contains classical conjunction.*

Proof. Classical conjunction has $\mathrm{F} \;\&\; \mathrm{F} = \mathrm{F}$; $\mathrm{F} \;\&\; \mathrm{T} = \mathrm{F}$; $\mathrm{T} \;\&\; \mathrm{F} = \mathrm{F}$; $\mathrm{T} \;\&\; \mathrm{T} = \mathrm{T}$. Equivalently paraconsistent conjunction has $-\infty \;\&\; -\infty = \min(-\infty, -\infty) = -\infty$; $-\infty \;\&\; \infty = \min(-\infty, \infty) = -\infty$; $\infty \;\&\; -\infty = \min(\infty, -\infty) = -\infty$; $\infty \;\&\; \infty = \min(\infty, \infty) = \infty$. □

Theorem 9. *Paraconsistent disjunction contains classical disjunction.*

Proof. Classical disjunction has $\mathrm{F} \vee \mathrm{F} = \mathrm{F}$; $\mathrm{F} \vee \mathrm{T} = \mathrm{T}$; $\mathrm{T} \vee \mathrm{F} = \mathrm{T}$; $\mathrm{T} \vee \mathrm{T} = \mathrm{T}$. Equivalently paraconsistent disjunction has $-\infty \vee -\infty = \max(-\infty, -\infty) = -\infty$; $-\infty \vee \infty = \max(-\infty, \infty) = \infty$; $\infty \vee -\infty = \max(\infty, -\infty) = \infty$; $\infty \vee \infty = \max(\infty, \infty) = \infty$. □

Theorem 10. *Paraconsistent negation, conjunction, and disjunction are well defined for all transreal arguments.*

Proof. Paraconsistent negation, conjunction, and disjunction are defined for all transreal arguments. It remains only to show that these operators are monotonic. Firstly nullity is absorptive in these operators so that if any argument is nullity the result is nullity. Nullity is disjoint from all other transreal numbers because it is the uniquely isolated point of transreal space [7], therefore nullity results are disjoint from all other transreal results and cannot contradict them. Secondly the preceding three theorems show that the paraconsistent operators are well defined at the boundaries $-\infty$ and ∞ but, by definition, the non-nullity, paraconsistent, truth values are monotonic so the operators just defined are monotonic for all transreal t in the range $-\infty \leq t \leq \infty$. This completes the proof for all transreal arguments. □

We now derive the paraconsistent negation, conjunction and disjunction from formulae involving the paraconsistent Sheffer Stroke. This proves that the paraconsistent Sheffer Stroke is functionally complete both for classical truth values and for the paraconsistent truth values defined here.

Theorem 11. *$pp| = \neg p$ for all transreal p.*

Proof. $pp| = \neg(p \;\&\; p) = \neg p$, with all symbols read paraconsistently. ☐

Theorem 12. *$pq|pq|| = p \;\&\; q$ for all transreal p, q.*

Proof. $pq|pq|| = (\neg(p \;\&\; q))(\neg(p \;\&\; q))| = \neg(\neg(p \;\&\; q)) = p \;\&\; q$, with all symbols read paraconsistently. ☐

Theorem 13. *$pp|qq|| = p \;\vee\; q$ for all transreal p, q.*

Proof. $pp|qq|| = (\neg p)\ (\neg q)\ | = \neg((\neg p) \;\&\; (\neg q)) = p \vee q$ by the classical de Morgan's Law, generalised to all transreal numbers by monotonicity and the absorptiveness of nullity, with all symbols read paraconsistently. ☐

2.4 Logical Space

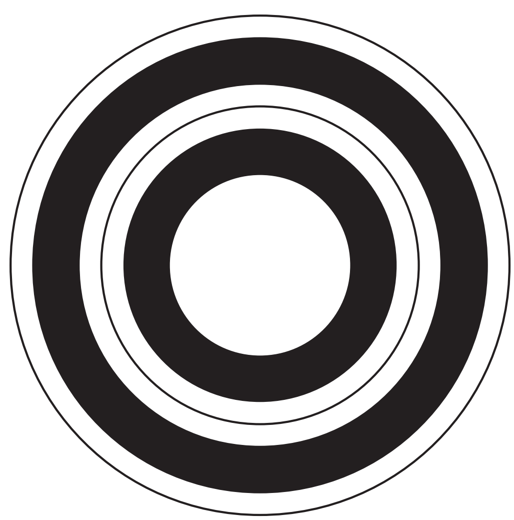

Figure 1: Two Bands of the Possible Worlds Rainbow

Wittgenstein discusses the concept of a logical space [16]. We now construct a transreal space of all paraconsistent, possible worlds whose predicates take on transreal truth values, including subsets such as real and Boolean truth values.

Firstly we note that the set of transreal numbers $\{t\ :\ -\infty \leq t \leq \infty,\ t = \Phi\}$ can be mapped onto, t', a line segment, with a length of one half of a unit, augmented by the point at nullity, using the transreal arctangent [7] as $t' = \arctan(t)/2\pi$. Thus all of the transreal, truth values for a single predicate can be mapped onto a half-unit line-segment, augmented with the point at nullity.

Secondly we list the countable infinitude of predicates, p_i, in some fixed order, with i running over the strictly positive integers. We then lay off the truth values of the ith predicate in a half-unit segment from i to $i + 1/2$, with the point at nullity at $i + 3/4$. A possible world has exactly one transreal, truth value or, equivalently, exactly one geometrical point, in the ith union of the ith line segment and ith point at nullity. Thus points on a half-infinite line encodes one possible world.

Thirdly we rotate the half-infinite line by a full rotation and index each radius by a real angle, θ, in the range $0 \leq \theta < 2\pi$. As each distinct radius is a distinct possible world, we have constructed a continuum of possible worlds whose cardinality is just sufficient to model the set of all possible worlds. To see the sufficiency of the cardinality note that we may select the truth values in one line segment with the cardinality of the continuum but the selections in the other line segments do not increase the cardinality [32].

The figure just constructed, see Figure 1, is a concentric, alternating sequence of an ith annulus and an ith circle, with an empty unit disc at the centre. We call the union of an annulus and its circle a “band” and we refer to the whole figure as a “rainbow” in anticipation of variants that are taken, not over a full rotation, but over an arc.

Many variations on this construction are possible. We might want to: let the angle range over more than a full rotation so that the windings encode a countable infinitude of partitions of all possible worlds; let the angle range over some quantity other than an integral number of full rotations so that both the lower and upper bounds of the range are included in the range; take the angle about zero so that it can be constructed by a Lie group [1]; take the angle in the range $-\pi/2 \leq \theta \leq \pi/2$ or $\theta = \Phi$ so that it is in the principal range of the transreal arctangent [7]. We might want to define some other planar shape entirely.

2.5 Partitioning, Significance and Consistency

One way to handle inconsistent predicates is to partition them into internally consistent partitions [26]. Da Costa [13] uses a hierarchy of such partitions. The partitions may be constructed syntactically or by semantic relevance [19][25][27]. Self evidently this produces islands of consistent reasoning but it does more. Priest [26] discusses the well known *preface paradox* in which an author writes a book containing many statements, which the author has good grounds to believe are true, but who also writes, in the preface, that in any list of many statements there must be errors. Thus the author is in the paradoxical, or paraconsistent, position of believing every single one of the statements and not believing, actually disbelieving, the conjunction of all of the statements.

This paradox is dissolved by Williams [33] who points out that a human can rationally believe a set of statements and not believe their conjunction. He cites the example of a motorcycle rider who knows a certain route that involves a left-hand bend followed by a right-hand bend. The rider successfully negotiates the route. Over time the rider comes to learn the conjunction that the left-hand bend is followed by the right-hand bend and changes behaviour to take the faster 'racing line' through the switchback. Thus it is human experience that one can know many propositions and not know their conjunction. Hence there is no mystery in the fact that a human can believe inconsistent predicates if he or she has not derived a contradiction. This issue is taken up in a description of how autoepistemic reasoning can be implemented in a finite (Prolog) implementation of a system's own beliefs [22]. Thus we have reason to expect that human and all finite reasoning may contain inconsistent statements and that one way to handle them is via partitioning.

There is a simpler way to handle inconsistencies using arithmetic. Suppose we have the sentence "p & $\neg p$." If we handle this sentence algebraically we may conclude, by conjunction elimination, that p is True and $\neg p$ is True, regardless of whether the truth values are classical or paraconsistent. But if we evaluate this proposition arithmetically we have either T & F = F or F & T = F, the conjunction fails, so we cannot derive a contradiction. Hence reasoning arithmetically blocks the derivation of contradictions but reasoning algebraically may admit them. It is no surprise, then, if human reasoning has evolved to be particular, working on concrete mental models, rather than having great facility in general reasoning [21].

Suppose we reason arithmetically and keep a record of the serial or parallel sequence of points we visit in a logical space. Then the trace is necessarily consistent because it was derived arithmetically. In a biological system, we might reinforce all of the neurones in the trace [14]; in a computer system we store the trace as a collection of line segments. In paraconsistent terms the trace is a partition. If we find more than one line segment passing through a point then all of the traces are consistent at that point and we may combine the common parts of the traces. The common parts form a partition that have been consistent over the entire history of the arithmetical reasoning system.

If all of the components on an axis in logical space are negative then all of the system's history supports the conclusion that the predicate tied to the axis is more False than True; if positive then the predicate is historically more True than False; if there are components that are both positive and negative then the predicate is historically contradictory or paraconsistent. Hence a reasoning system that has sufficient geometrical or algebraic power may detect the inconsistency or paraconsistency. If many traces haven components in this axis then a contradiction or paraconsistency is important to the mental life of the system. This bears on a case discussed by Priest [26]. Bohr's model of the atom is useful in predicting atomic structure and is, in our terms, associated with many traces. Maxwell's equations are useful in electrodynamics and are also associated with many traces. But the two are inconsistent in that the orbit of a Bohr electron should decay due to Maxwell radiation, which it does not. The non-decay is therefore significant, because it is associated with many traces, and finds resolution in the combination of quantal energy levels and the Pauli exclusion principle. Thus logical space offers a measure of the significance of a predicate, in terms of the number of traces that pass through it in logical space, and we now arrive at a measure which classifies classical and dialetheaic consistencies. Let our desideratum be $d = p \;\&\; \neg p$. If $d = -\infty$ then predicate p is a classical consistency; if $-\infty < d < 0$ then p is a dialetheaic consistency; if $d = 0$ then p is both a dialetheaic consistency and a dialetheaic contradiction; if $0 < d < \infty$ then p is dialetheaic contradiction; if $d = \infty$ then p is a classical contradiction; if $d = \Phi$ then p is a Gap.

2.6 Advanced Paraconsistent Logics

We have deliberately introduced an elementary paraconsistent logic. It is just strong enough to generalise propositional logic [23] to paraconsistent form but this means it is a basis for generalising more advanced paraconsistent logics. For example the computer language, Prolog [12], provides a propositional logic [23] with existential and universal quantifies introduced by controlling the scope of variables. Variables with scope local to a clause are existential, while variables with global scope are universal [31] (p. 10). Any propositional logic is trivially extended by the paraconsistent logic presented here but the example of Prolog raises a deeper issue. Prolog works by unifying variables with arguments. The unification triggers execution of the clause but in a paraconsistent logic, such as ours, all paraconsistent truth values will unify with arguments *to some extent.* One way to handle this would be to allow all clauses to execute in parallel but it is unlikely that a practical machine would have the resources to do this for many clauses. A more practical approach might be to allow only the execution of clauses that have a high degree of truthfulness, embedding a best-first strategy in the execution of the computer language. Triggering on truthfulness reflects the human bias for positive reasoning but in an artificial intelligence we may prefer to trigger on a high degree of divergence from being equally False and True. In a practical system we might want some measure of usefulness, expected value [17], or subjective value [14] to drive the selection of clauses, in which case we could couple a value parameter with clauses. This takes us deeply into the design of robots and the modelling of animal and artificial intelligences.

The matching of all paraconsistent truth values to a clause's variables leads us to expect that useful paraconsistent logics will have some implicit or explicit notion of execution control, raising the prospect that they should be as powerful as the Turing machine. Clearly paraconsistent logics can be very powerful, taking them a very long way from the elementary, paraconsistent logic introduced here. Our paraconsistent logic may be of interest precisely because of its mathematical simplicity. We invite the reader to consider if more competent, paraconsistent logics, would benefit from being arithmetised in transreal arithmetic rather than calling on the advanced, even heroic, mathematics that is usually used as a consistent basis for a paraconsistent logic.

3 Conclusion

We present a paraconsistent logic by arithmetising the Sheffer Stroke in transreal arithmetic. We use negative infinity to model classical False and positive infinity to model classical True. The real numbers model truth values that are both False and True. The negative, real numbers are more False than True; the positive, real numbers are more True than False; zero models the truth value that is equally False and True. The transreal number nullity, which is unordered and lies outside the range of ordered numbers from negative infinity to positive infinity, models truth values that have no component of falsity or truthfulness. These *Gap* truth values are neither False nor True. As the Sheffer Stroke can be used to develop all classical, truth functional logics, our logic is just one of a class of paraconsistent generalisations of classical logic. The reader, who is skilled in formal logic, can easily develop other paraconsistent logics using transreal arithmetic as a consistent basis. This is a very much simpler approach to the formalisation of paraconsistent logics than is usually attempted so it brings paraconsistent logics within the reach of non-specialists, such as Electronic Engineers and Computer Scientists, who have been trained in classical, Boolean logic.

The Sheffer Stroke directly implements the operation Not-And (NAND). NAND gates can be used to implement all of the logic in general purpose computers so our paraconsistent generalisation of the Sheffer Stroke specifies the behaviour of a paraconsistent NAND gate. This could be used to implement paraconsistent processors in hardware. Such hardware might be particularly well suited to the execution of Artificial Intelligence and Cybernetics programs, for example those which exploit models of neural processing or fuzzy logic.

We give a geometrical construction which encodes all paraconsistent, possible worlds in an alternating sequence of an annulus and a circle that fills out the real plane, with each radius marking off one possible world.

We point out the obvious property that evaluation of logical sentences involving only connectives and literal False and True values, not variables, cannot support the conclusion of a contradiction. Thus on-the-fly evaluation of a literal sentence is always consistent. We propose that this property can be used to partition predicates into internally consistent sets and that this provides a measure of the significance of predicates. We also give a simple, transarithmetical measure, a desideratum, $d = p \,\&\, \neg p$, for categorising predicates into classical or else dialetheaic form and for categorising classical or else dialetheaic consistencies and contradictions.

We give a very simple principle of metalogical monotonicity that forces a logic to be non-explosive. In future we might examine existing paraconsistent logics to see if they implicitly obey this principle.

Given the interest there is in the number nullity, perhaps the single most important, original contribution of this paper is to observe that the transreal number nullity provides a faithful model of Gap values in advanced logics.

References

[1] Simon L. Altmann. *Rotations, Quaternions and Double Groups.* Dover Publications Inc, 1986.

[2] J. A. D. W. Anderson. Representing geometrical knowledge. *Phil. Trans. Roy. Soc. Lond. Series B.*, 352(1358):1129–1139, 1997.

[3] James A. D. W. Anderson. Exact numerical computation of the rational general linear transformations. In Longin Jan Lateki, David M. Mount, and Angela Y. Wu, editors, *Vision Geometry XI*, volume 4794 of *Proceedings of SPIE*, pages 22–28, 2002.

[4] James A. D. W. Anderson. Perspex machine iii: Continuity over the turing operations. In Longin Jan Lateki, David M. Mount, and Angela Y. Wu, editors, *Vision Geometry XIII*, volume 5675 of *Proceedings of SPIE*, pages 112–123, 2005.

[5] James A. D. W. Anderson. Evolutionary and revolutionary effects of transcomputation. In *2nd IMA Conference on Mathematics in Defence.* Institute of Mathematics and its Applications, Oct. 2011.

[6] James A. D. W. Anderson. Trans-floating-point arithmetic removes nine quadrillion redundancies from 64-bit ieee 754 floating-point arithmetic. In *this present proceedings*, 2014.

[7] James A. D. W. Anderson and Tiago S. dos Reis. Transreal limits expose category errors in ieee 754 floating-point arithmetic and in mathematics. In *Submitted for consideration in this proceedings*, 2014.

[8] James A. D. W. Anderson, Norbert Völker, and Andrew A. Adams. Perspex machine viii: Axioms of transreal arithmetic. In Longin Jan Lateki, David M. Mount, and Angela Y. Wu, editors, *Vision Geometry XV*, volume 6499 of *Proceedings of SPIE*, pages 2.1–2.12, 2007.

[9] Robert Audi. *The Cambridge Dictionary of Philosophy.* Cambridge University Press, 2nd edition, 1997.

[10] R. E. Bellman and L. A. Zadeh. Local and fuzzy logics. In J. M. Dunn and G. Epstein, editors, *Modern Uses of Multiple-Valued Logic*, pages 103–165. D. Reidel Publishing Co., 1977.

[11] W. Carnelli and J. Marcos. Ex contradictione non sequitur quodlibet. In *Proc. 2nd Conf. on Reasoning and Logic*, Bucharest, 2001.

[12] W. F. Clocksin and C. S. Mellish. *Programming in Prolog.* Springer-Verlag, 1981.

[13] N. C. A. da Costa. The philosophical import of paraconsistent logic. *Journal of Non-Classical Logic*, 1:1–19, 1982.

[14] Gerald M. Edelman and Giulio Tononi. *A Universeo of Conscousness - How Matter Becomes Imagination.* Basic Books, 2000.

[15] William Feller. *An Introduction to Probability Theory and its Applications*, volume 1 of *Probability and Mathematical Statistics*. Wiley, 3rd edition, 1957.

[16] Juliet Floyd. Wittgenstein on philosophy of logic and mathematics. In Stewart Shapiro, editor, *Oxford Handbook of Philosophy of Logic and Mathematics*, chapter 4, pages 75–128. Oxford University Press, 2005.

[17] John E. Freund and Ronald E. Walpole. *Mathematical Statistics.* Prentice Hall, 3rd edition, 1980.

[18] Walter Gomide. O princípio de não-contradição e sua tradução para a aritmética transreal. *Investigação Filosófica*, 4(1), 2013.

[19] Russell Greiner and Michael R. Genesereth. What's new? a semantic definition of novelty. In *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pages 450–454. William Kaufmann, 1983.

[20] Theo M.V. Jansen. Frege, contextuality and composionality. *Journal of Logic, language, and Information*, 10, 2001.

[21] Philip Johnson-Laird. *Mental Models: Towards a Cognitive Science of Language, Inference and Consciousness.* Cambridge University Press, 1983.

[22] Robert C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25(1):75–94, Jan 1985.

[23] John Nolt and Dennis Rohatyn. *Logic.* McGraw-Hill, 1988.

[24] G. Priest, J. C. Beall, and B. A. Gab. *The Law of Non-Contradiction.* Oxford University Press, 2004.

[25] Graham Priest. *An Introduction to Non-Classica Logic From If to Is.* Cambridge University Press, 2nd edition, 2008.

[26] Graham Priest, Koji Tanaka, and Zach Weber. Paraconsistent logic. *The Stanford Encyclopedia of Philosophy*, http://plato.stanford.edu/archives/fall2013/entries/logic-paraconsistent, 2013.

[27] Stephen Read. *Relevant Logic - A Philosophical Examination of Inference.* Basil Blackwell, 2012 (first published 1988).

[28] Kenneth H. Rosen. *Discrete Mathematics and its Appications.* McGraw-Hill, 6th edition, 2007.

[29] Y. Shramko and H. Wansing. Entailment relations and truth values. *Bulletin of the Section of Logic*, 36:131–143, 2007.

[30] Y. Shramko and H. Wansing. Truth and falsehood. an inquiry into generalised logical values. *Trends in Logic*, 36, 2007.

[31] Leon Sterling. *The Art of Prolog - Advanced Programming Techniques.* MIT Press, 1986.

[32] S. Swierczkowski. *Sets and Numbers.* Routledge & Keegan Paul, 1972.

[33] John N. Williams. The preface paradox resolved. *Theoria*, 2(3):121–140, 1987.

[34] L. A. Zadeh. Fuzzy logic and approximate reasoning. *Synthese*, 30:407–428, 1975.