Temporal Partial Metric Spaces
The Distance from Fréchet to Google

Steve Matthews
Department of Computer Science
University of Warwick
Coventry, CV4 7AL, UK
Steve.Matthews@warwick.ac.uk

Anatolian Communications in NonLinear Analysis
Abant ˙Izzet Baysal Üniversitesi, July 3rd-6th 2013

Abstract
Partial metric spaces generalise Fréchet’s metric spaces by allowing self-distance to be a positive number. Originally motivated by the goal to reconcile metric space topology with the logic of computable functions and Dana Scott’s innovative theory of topological domains they are now too static a form of mathematics to be of use in modelling contemporary application software (aka Apps) which is increasingly pragmatic, interactive, and inconsistent, in nature. This talk addresses the reality that if partial metric spaces are to survive in future research then they must firstly progress by means of a computable temporal generalisation. How can this be achieved? (1) The induced partial ordering is temporally ordered in the sense of the Lucid algebra, (2) time is quantifiable in the sense of the time complexity of algorithms, and (3) time can be paused in the sense of Wadge’s hiatus.

1 Metric spaces and computing
1.1 Necessary zero self-distance

"You cannot be serious!
John McEnroe, mad on Centre Court, Wimbledon 1981

1 Metric spaces and computing
1.1 Necessary zero self-distance

As with most mathematics there is an unquestioned philosophical identification in the theory of metric spaces between ontology and epistemology. That is, all that exists in a metric space is presumed knowable (by examining distances), and all that is knowable (by distances) is presumed to exist.

Hence in the time of Fréchet it was eminently sensible to axiomatize self-distance by \(d(x,x) = 0\) and \(d(x,y) = 0 \Rightarrow x = y\).

Example (1)
\[
\begin{align*}
|\cdot - | : (-\infty, +\infty)^2 & \rightarrow [0, \infty)\quad \text{where} \quad |x - y| = x - y \quad \text{if} \quad y \leq x, \quad y - x \quad \text{otherwise.}
\end{align*}
\]

Example (1)
\[
\begin{align*}
\text{A metric space} \quad \text{(Fréchet, 1906)} & \quad \text{is a tuple} \\
(X, d : X \times X & \rightarrow [0, \infty)) \quad \text{such that,}
\end{align*}
\]

\[
\begin{align*}
d(x,x) & = 0 \\
d(x,y) & = 0 \Rightarrow x = y \\
d(x,y) & = d(y,x) \\
d(x,z) & \leq d(x,y) + d(y,z)
\end{align*}
\]

Example (1)
1 Metric spaces and computing
1.1 Necessary zero self-distance

Definition (2)
For each metric space \((X, d)\) a sequence \(x_0, x_1, \ldots\) is \(\text{Cauchy}\) if for each \(\epsilon > 0\) there exists \(N\) s.t. \(n, m \geq N \Rightarrow d(x_n, x_m) < \epsilon\).

Definition (3)
For each metric space \((X, d)\) a sequence \(x_n\) \(\text{converges}\) to a point \(x \in X\) if for any \(\epsilon > 0\) there exists \(N\) s.t. \(d(x_n, x) < \epsilon\) for any \(n \geq N\).

Definition (4)
A metric space \((X, d)\) is \(\text{complete}\) if every Cauchy sequence converges.

Definition (5)
For each metric space \((X, d)\) and function \(f : X \rightarrow X\), \(f\) is a \(\text{contraction (mapping)}\) if there exists a constant \(0 < c < 1\) s.t. \(d(f(x), f(y)) \leq c \times d(x, y)\) for all \(x, y \in X\).

Definition (6)
For each metric space \((X, d)\) and function \(f : X \rightarrow X\), \(a \in X\) is a \(\text{fixed point}\) of \(f\) if \(f(a) = a\).

Theorem (1) (Banach, 1922)
Each contraction over a complete metric space has a (unique) fixed point.

Stefan Banach (1892-1945)

1.2 Non-zero self-distance

Example (3)
Let \(d\) be the metric on the set \([F, T]\) of truth values \(false\) \& \(true\) such that \(d(F, T) = 1\). Then \(T_d = 2^{\{F, T\}}\), and \(\{\{F\}, \{T\}\}\) is a basis for \(T_d\).

1.2 Non-zero self-distance

- The past century has taught us that some problems are \(\text{undecidable}\) (i.e. there are more \(\text{truths}\) in the world than are sound reasoning of \(\text{proofs}\)).
- More recently \(\text{fallacies}\) (i.e. presently believed truths but subsequently proved falsehoods) such as might arise in large scale data processing are becoming unavoidable in practice.
- How well prepared are the mathematical certainties of metric spaces for today’s information processing age? How can we incorporate the cost of deriving mathematical truths? How can we incorporate ever more unsound reasoning into Computer Science?
1 Metric spaces and computing

1.2 Non-zero self-distance

The ideal of logic for ascertaining perfect truth in Star Trek’s portrayal of the 23rd century is a wonderful caricature of how we appreciate undecidable problems and fallacies in the information age of the 20-21st centuries.

Captain Kirk arbitrates between his closest confidants, the totally logical Mr Spock and the passionate Dr McCoy.

Mr Spock’s reasoning is infallible but as a result incomplete. Dr McCoy’s reasoning is emotional, daring, wider reaching, but as a result fallible.

Just as Kirk needs Spock and McCoy to explore the galaxy so we need to bring together the infallibility of logical reasoning with the abstract expressiveness of (say) topology.

Hausdorff separability was taken for granted until the 20th century story of Bletchley Park by Alan Turing, with an important refinement devised in 1940 by Gordon Welchman. The engineering design and construction was the work of Harold Keen of the British Tabulating Machine Company. ...

Russell in 1916

Russell’s paradox in point set topology by neighbourhoods is approximation of a totally known limit point by totally known open sets.

So, is not an open set such as an open ball $B_r(a)$ a sufficient notion of approximation for each and every point $x \in B_r(a)$? The answer would be YES! if ontology and epistemology of naive set theory were consistent & synonymous.

Russell’s paradox of 1901 argues that if we could define $ R = \{x | x \notin x\}$ then $R \in R \iff R \notin R$.

This paradox (i.e. self contradiction) known to Russell et.al. led to our understanding of incompleteness in logic and computability theory.

“...
1 Metric spaces and computing

1.2 Non-zero self-distance

Definition (11)

A topological space \((X, \tau \subseteq 2^X)\) is **T₀ separable** if

\[ a \neq b \Rightarrow \exists O \in \tau . (a \in O \land b \notin O) \lor (b \in O \land a \not\in O) \]

(distinct points can be separated by a neighbourhood)

Lemma (2)

\(T_2 \Rightarrow T_0\)

Inspired by Dana Scott in the 1960s, \(T_0\) separable spaces became a topological model for the logic of computable functions.

Definition (12)

The **information ordering** (in the study of consistent approximation) is,

\[ a \sqsubseteq b \iff (\forall O \in \tau . a \in O \Rightarrow b \in O) \]

“... It was a substantial development from a device that had been designed in 1938 by Polish Cipher Bureau cryptologist Marian Rejewski, and known as the "cryptologic bomb" (Polish: "bomba kryptologiczna"). The bombe was designed to discover some of the daily settings of the Enigma machines on the various German military networks: specifically, the set of rotors in use and their positions in the machine; the rotor core start positions for the message, the message key, and one of the wirings of the plubboard.”

(Wikipedia)

Machine Room, Hut 6, 1943

Enigma machine

1.3 Perspectives upon partial metric spaces

A difficult lesson has been that I cannot begin a research talk such as this with a formal definition of partial metric space and expect to be able to derive & communicate all it’s interesting properties. First we need to appreciate that there are many legitimate perspectives.

1.3.1 Strong mathematics

1.3.2 Computability

1.3.3 Falling between two stools

1.3.4 Distance has to be more cost-sensitive

A weighted metric space is a tuple

\((X, d, |·| : X \rightarrow (−\infty, \infty))\) such that \((X, d)\) is a metric space.

1 Metric spaces and computing

1.2 Non-zero self-distance

Definition (11)

A topological space \((X, \tau \subseteq 2^X)\) is **T₀ separable** if

\[ a \neq b \Rightarrow \exists O \in \tau . (a \in O \land b \notin O) \lor (b \in O \land a \not\in O) \]

(distinct points can be separated by a neighbourhood)

Lemma (2)

\(T_2 \Rightarrow T_0\)

Inspired by Dana Scott in the 1960s, \(T_0\) separable spaces became a topological model for the logic of computable functions.

Definition (12)

The **information ordering** (in the study of consistent approximation) is,

\[ a \sqsubseteq b \iff (\forall O \in \tau . a \in O \Rightarrow b \in O) \]

“... It was a substantial development from a device that had been designed in 1938 by Polish Cipher Bureau cryptologist Marian Rejewski, and known as the "cryptologic bomb" (Polish: "bomba kryptologiczna"). The bombe was designed to discover some of the daily settings of the Enigma machines on the various German military networks: specifically, the set of rotors in use and their positions in the machine; the rotor core start positions for the message, the message key, and one of the wirings of the plubboard.”

(Wikipedia)

Machine Room, Hut 6, 1943

Enigma machine

1.3 Perspectives upon partial metric spaces

A difficult lesson has been that I cannot begin a research talk such as this with a formal definition of partial metric space and expect to be able to derive & communicate all it’s interesting properties. First we need to appreciate that there are many legitimate perspectives.

1.3.1 Strong mathematics

1.3.2 Computability

1.3.3 Falling between two stools

1.3.4 Distance has to be more cost-sensitive

A weighted metric space is a tuple

\((X, d, |·| : X \rightarrow (−\infty, \infty))\) such that \((X, d)\) is a metric space.
1 Metric spaces and computing
1.3 Perspectives upon partial metric spaces

1.3.2 Computability: in contrast the pre-Computer Science perspective (from logic & set theory) of partial metric spaces is that metric spaces & general topology can be, and indeed must be for the sake of realism, meaningfully combined with existing research into the logic of computable functions. Thus (1969) Dana Scott’s groundbreaking $T_0$-separable domain theory model for computable functions in mathematics. Sufficient for the $x \subseteq y$ information ordering of computability theory, but a one-way street insufficient for today’s world of Apps & tablet computers.

2 Failure takes time
2.2 Cost-free 2, 3, & 4 valued truth logic

⊤ (pronounced top in lattice theory and both in four valued logic) introduces the possibility of an overdetermined mathematical value, $F, T$ (pronounced false, true) typifies two well defined consistent distinct values in a mathematical theory, ⊥ (pronounced bottom in lattice theory and either in four valued logic) introduces the possibility of an underdetermined mathematical value.

1.3.3 Falling between two stools: the strong Math respect for metric spaces identified with (Hausdorff) $T_2$-separability understandably can see no merit in the weaker $T_0$-separability of partial metric spaces. The computability perspective in common with domain theory in general has not developed in tandem with the self-evident progress of research into applied computing to be of service therein. The result is that partial metric spaces have deservedly fallen between the two stools of the two perspectives. That is, partial metric spaces have failed to be a significant development of either metric spaces or of domain theory. Similarly, when considered in combination, partial metric spaces have failed to advance in tandem with the multi-disciplinary lead of applied computing.

Hence the term failure takes time (given to the speaker by W.W. Wadge) is used here to reconcile standard approaches to introducing time & cost in computation. In the pre-computing cost-free world of two-valued truth logic we can take for granted the property $P \iff \neg\neg P$. Later in logic programming, still a two-valued logic, a term not $P$ is assumed to hold (rightly or wrongly) from the so-called failure to prove (by computable means) the truth of $P$. The following example is adapted from a Prolog manual.

unmarried_student(X) :- not(married(X)), student(X)
married(john)

unmarried_student(joe) holds as not(married(joe)) fails.

as computing power and apps grow so does the scope for humans to enter and expect computers to (consistently) process their inconsistent data. $F, T$ the traditional idealised realm of mathematics in which all values, axioms, theorems, etc. are presumed to be logically consistent.

perhaps the wait is finite, perhaps infinite, we just cannot know how long if ever it will take to compute the desired mathematical value.
2 Failure takes time

2.2 Cost-free 2, 3, & 4 valued truth logic

We assume that information is partially ordered.
\[ \bot \subseteq F \subseteq T \text{ and } \bot \subseteq T \subseteq T. \]
\[ F \not\subseteq T \text{ and } T \not\subseteq F \text{ (two valued logic is unchanged).} \]
x \subseteq y \Rightarrow f(x) \subseteq f(y) \text{ (functions are monotonic).}

We envisage a vertical structure of information processing orthogonal to a given horizontal mathematical structure.

Definition (14)
A partially ordered set (poset) is a relation \((X, \subseteq X \times X)\) such that
\[ x \subseteq x \text{ (reflexivity)} \]
\[ x \subseteq y \land y \subseteq x \Rightarrow x = y \text{ (antisymmetry)} \]
\[ x \subseteq y \land y \subseteq z \Rightarrow x \subseteq z \text{ (transitivity)} \]

Definition (15)
For each poset \((X, \subseteq\), \((X, \subseteq)\) is the relation such that,
x \subseteq y \Leftrightarrow x \subseteq y \land x \neq y.

Example (4)
The real numbers \((-\infty, \infty)\) are partially ordered by the relation
\[ x \subseteq y \text{ iff } x \geq y. \]

Definition (16)
A lattice is a partially ordered set in which each pair of points \(x, y\) has a unique greatest lower bound (aka \textit{infinum} or \textit{meet}) denoted \(x \cap y\), and unique lowest upper bound (aka \textit{supremum} or \textit{join}) denoted \(x \cup y\).

Example (5)
A set is a lattice when partially ordered by set inclusion. Infinum is set intersection, and supremum is set union.

Example (6)
The extension of two valued truth logic from \(\{F, T\}\) to \(\{F, \bot, \top, T\}\) where \(F \cap T = \top\) and \(F \cup T = \bot\) is a lattice.

Definition (17)
A two argument function (in infix notation) \(op\) is symmetric if \(x \, op \, y = y \, op \, x\) for all \(x\) and \(y\).

Definition (18)
A two argument function (in infix notation) \(op\) is associative if \(x \, op \, (y \, op \, z) = (x \, op \, y) \, op \, z\) for all \(x, y, \& z\).

Example (7)
In a distributive lattice, \textit{meet} & \textit{join} exist, are symmetric, are associative, and \(x \cup (y \cap z) = (x \cup y) \cap (x \cup z)\) and \(x \cap (y \cup z) = (x \cap y) \cup (x \cap z)\).

Definition (19)
A function \(f\) over a lattice is distributive if \(f(x \cap y) = (f(x)) \cap (f(y))\) and \(f(x \cup y) = (f(x)) \cup (f(y))\).

Truth table for sequential and (computing left-to-right).

\[
\begin{array}{c|c|c|c|c}
P \land Q & F & T & F & T \\
\hline
F & F & F & F & F \\
T & F & T & F & T \\
\end{array}
\]

Sequential and is monotonic, not symmetric as \(\bot \land F \neq F \land \bot\), and \(\bot \land \bot = \bot\) for each \(Q\).

Truth table for parallel and (Bolcan logic).

\[
\begin{array}{c|c|c|c|c|c|c}
P \lor Q & F & F & F & F & F & F \\
\hline
F & F & F & F & F & F & F \\
T & F & T & F & T & F & T \\
\end{array}
\]

Parallel and is monotonic, symmetric, distributive, and above sequential and.
2 Failure takes time
2.2 Cost-free 2, 3, & 4 valued truth logic

▶ And so we see that the structure of two valued truth logic \( (F, \top) \) can be generalised in such a way that renders it more relevant to incorporate concepts which become both obvious and necessary in the age of computing.

▶ It is, in effect, sufficient to use our intuition to extend two valued truth logic. Sadly it is more challenging to see how to extend more sophisticated mathematical structures such as metric spaces.

▶ The insight here is to appreciate that the metric concept of self-distance could meaningfully (as in each of mathematics and CS) be non-zero as well as zero.

2 Failure takes time
2.3 A cost for computing negation

Definition (20)
A partial metric space (Matthews, 1992) is a tuple \( (X, \rho : X \times X \rightarrow [0, \infty]) \) such that,
\[
\rho(x, x) \leq \rho(x, y) \quad (\text{small self-distance})
\]
\[
\rho(x, x) = \rho(y, y) = \rho(x, y) \Rightarrow x = y \quad (\text{equality})
\]
\[
\rho(x, y) = \rho(y, x) \quad (\text{symmetry})
\]
\[
\rho(x, z) \leq \rho(x, y) + \rho(y, z) - \rho(y, y) \quad (\text{trianglearity})
\]

Lemma (21)
An open ball \( B_{\epsilon}(a) = \{ x \in A : \rho(a, x) < \epsilon \} \).

Lemma (3)
The open balls form the basis for a topology \( \tau_{0} \). This is asymmetric in the sense that there may be \( x, y \) such that \( y \in cl\{x\} \land x \notin cl\{y\} \) (i.e. \( T_{0} \) separation).

Note that a metric space \( (X, \rho) \) is not linear in the sense that the function \( d'(x, y) = a \times d(x, y) + b \) for arbitrary \( a > 0 \), \( b \geq 0 \) is not in general a metric; this being because of the necessary zero self-distance axiom \( d(x, x) = 0 \).

In contrast a partial metric space \( (X, \rho) \) is linear as the function \( \rho'(x, y) = a \times \rho(x, y) + b \) is also a partial metric, and furthermore \( \tau_{0}' = \tau_{0} \).

Thus now we can generalise \( (X, \rho : X \times X \rightarrow [0, \infty]) \) to \( (X, \rho : X \times X \rightarrow (-\infty, \infty)) \) to express negative distances.
2 Failure takes time
2.4 Fixed points and partial metric spaces

\[ F \rightarrow \top \]
\[ \bot \]

In the least fixed point domain theory tradition of Kleene, Alfred Tarski (1926), & Scott \( \neg T = F \), \( \neg F = T \), \( \neg \bot = \top \), \( \neg \top = \bot \), and \( \bigwedge_{n \geq 0} \neg^n(\bot) \) can be used to define the ideal meaning for \( \psi \) recursively defined by \( \psi = \neg \psi \).

The problem in 1980 for myself was how to reconcile the Banach contraction mapping theorem of metric spaces with Tarski’s least fixed point theorem of chain complete posets.

Definition (22 (Matthews, 1992))
For each partial metric space \((X, p)\), and for each \(x \in \omega \rightarrow X\), \(x\) is a Cauchy sequence if,
\[
\forall \epsilon > 0 \ \exists k \in \omega \ \forall n, m > k \ . \ p(x_n, x_m) < \epsilon
\]

Definition (23 (Matthews, 1992))
A partial metric space is complete if for each Cauchy sequence \(x \in \omega \rightarrow X\) there exists \(a \in X\) s.t.,
\[
\lim_{n \rightarrow \infty} p(x_n, a) = 0
\]

Note: these definitions (and the fixed point theorem below) work for non-negative valued partial metric spaces.
We have yet to find a way to generalise them further.

2 Failure takes time
2.5 The cost of computing knowledge

Jeff Crouse “Recursion shirt”

Defining \( \psi \) to be \( \neg \psi \) is \( \psi = \bot \) in Tarski’s ideal world of least fixed points, but NOT in the real world of today’s computing.

Any attempt to compute the value of \( \psi \) up from \( \bot \) would sooner or later time out with an error message such as Control Stack Overflow.

Nick D’Aloisio (1995-)

“Longer term this whole journey has given me an appetite for starting companies, so I’d love to do it again ...” he told Sky News (26/03/13).
2 Failure takes time
2.5 The cost of computing knowledge

Dana Scott’s inspired work has given us the T₀ topology to model partial information. But, a computable function could still be an impossible object as depicted in Escher’s Waterfall which appears to produce an unending flow of water.

The supposed paradox disappears in our temporal partial metric spaces where (dynamic) cost can be composed with (static) data content.

Waterfall
Lithograph by M.C. Escher (1961)

Let us now consider temporal 3 valued propositional logic.

▶ Introducing the notion of cost to computing a partial metric distance \( p(x, y) \) is an important initial step in developing a future theory of metric spaces that may be termed dynamic, interactive, adaptive, or intelligent.

▶ For example, in classical logic we want to retain the double negative elimination theorem \( \neg\neg P \iff P \) while in addition asserting that \( \neg\neg A \) is more costly to compute than \( A \).

▶ Applying this idea to our earlier definition \( \psi = \neg\neg\psi \) would mean that we could retain the ideal Kleene, Tarski, & Scott domain theory meaning \( \sqcup_{n \geq 0} \neg^n(\bot) \) of least fixed points and use cost as a criteria for defining an error Control Stack Overflow.

In temporal three-valued propositional logic we can now define the following temporal truth table for negation.

<table>
<thead>
<tr>
<th>( \neg P )</th>
<th>( \bot )</th>
<th>( T )</th>
<th>( F )</th>
<th>( )</th>
<th>( \text{argument value} )</th>
<th>( \text{function definition} )</th>
<th>( \text{cost of a data value} )</th>
<th>( \text{cost of an hiatus} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P )</td>
<td>( c )</td>
<td>( c )</td>
<td>( c )</td>
<td>( c )</td>
<td>( )</td>
<td>( )</td>
<td>( )</td>
<td>( )</td>
</tr>
<tr>
<td>( (P \circ)_c )</td>
<td>( P_{c+1} )</td>
<td>( P_{c+1} )</td>
<td>( P_{c+1} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From this table it might seem odd that \( T_c = c \) & \( F_c = c \) for any \( c \geq 0 \) as surely each is a constant value needing only constant time to become known by means of computation. True, but one can also reasonably argue that once computed there is a persistent cost in computing to retaining a representation of each data value in memory.
3 There's no such thing as a free lunch
3.1 Wadge's hiatus for computing

- Wadge is much respected for his PhD UC Berkeley research known as the Wadge hierarchy, levels of complexity for sets of reals in descriptive set theory.

- Wadge's later insight that a complete object is "one that cannot be further completed" led from metric spaces (of complete objects), to Lucid (for programming over metric spaces), to partial metric spaces (domain theory for metric spaces), and now to discrete partial metric spaces (complexity theory for domains).

- "I don't know if infinitesimal logic is the best idea I've ever had, but it's definitely the best name. So here's the idea: a multivalued logic in which there are truth values that are not nearly as true as 'standard' truth, and others that are not nearly as false as 'standard' falsity" (Bill Wadge's blog, 3/2/11).

In 1977 Ashcroft & Wadge introduced a functional programming language called Lucid in which each input (resp. output) is a finite or infinite sequence of data values termed a history.

In domain theory parllance,

\[ \{\} \sqsubseteq \langle a \rangle \sqsubseteq \langle a, b \rangle \sqsubseteq \langle a, b, c \rangle \sqsubseteq \langle a, b, c, d \rangle \sqsubseteq \ldots \]

where the totally defined inputs are precisely the infinite sequences, and the partially defined inputs are precisely the finite sequences.

- Subsequently in partial metric space terms \( p(x, y) = 2^{-n} \) where \( n \) is the largest integer (or \( \infty \) if \( x = y \) is an infinite sequence) such that for each \( 0 \leq i < n \) \( x_i = y_i \). Then \( p \) is a partial metric inducing the above required ordering.

Wadge appreciated that "When you look for long into an abyss, the abyss also looks into you" (attributed to Friedrich Nietzsche).

"When we discovered the dataflow interpretation of Lucid (see post, Lucid the dataflow language) we thought we'd found the promised land. We had an attractive computing model that was nevertheless faithful to the declarative semantics of statements-as-equations. However, there was a snake in paradise, as David May explained in the fateful meeting in the Warwick Arts Center Cafeteria. ... And it's the need to discard data that leads to serious trouble. It could be that a huge amount of resources were required to produce ... resources that were wasted because we threw it out. But a real catastrophe happens if the data to be discarded requires infinite resources. Then we wait forever for a result that is irrelevant, and the computation deadlocks." (Bill Wadge's blog, 6/12/11)

But \( * \) is neither a well defined null data value (such as is the number 0 ) nor is say \( \langle *, 2 \rangle \) a partial data value comparable to \( \langle 2 \rangle \) in the partial ordering of domain theory. And so, what is a hiaton ?

The following table is an example history of how pauses could be envisaged as Lucid sequences.

<table>
<thead>
<tr>
<th>Time point</th>
<th>Scott domain ordering</th>
<th>Hiaton intuition</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(1)</td>
<td>(1)</td>
</tr>
<tr>
<td>1</td>
<td>(1)</td>
<td>(1, +)</td>
</tr>
<tr>
<td>2</td>
<td>(1, 2)</td>
<td>(1, +, 2)</td>
</tr>
<tr>
<td>3</td>
<td>(1, 2, 3)</td>
<td>(1, +, 2, +)</td>
</tr>
<tr>
<td>4</td>
<td>(1, 2, 3)</td>
<td>(1, +, 2, +, 3)</td>
</tr>
<tr>
<td>5</td>
<td>(1, 2, 3)</td>
<td>(1, +, 2, +, 3, +)</td>
</tr>
<tr>
<td>6</td>
<td>(1, 2, 3)</td>
<td>(1, +, 2, +, 3, +)</td>
</tr>
<tr>
<td>7</td>
<td>(1, 2, 3)</td>
<td>(1, +, 2, +, 3, +)</td>
</tr>
</tbody>
</table>

| ...        | ...                   | ...              |

Sadly in 1977 this correct insight did not have the mathematics to back it up!
3 There's no such thing as a free lunch
3.2 Temporal partial metric spaces

Definition (24)

Temporal partial metric space is a function

\[ p : ω \to (X \times X \to [0, \infty]) \]

such that each \( (X, p_t) \) is a partial metric space, and \( p_t(x, y) > p_{t+1}(x, y) \)

\[ p_t(x, y) \]

is read as the distance at time \( t \) between \( x \) and \( y \).

We reject the hiatus as being a data value in favour of a dynamically changing history of partial metric spaces.

### Table: Temporal partial metric spaces

<table>
<thead>
<tr>
<th>Time</th>
<th>Scott ordering</th>
<th>Hiaton intution</th>
<th>Temporal partial metric self distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>{}</td>
<td>{}</td>
<td>2^{-1} + (2^{-3} - 2^{-1}) \times 2^{-0}</td>
</tr>
<tr>
<td>1</td>
<td>(1)</td>
<td>(1)</td>
<td>2^{-1} + (2^{-3} - 2^{-1}) \times 2^{-0}</td>
</tr>
<tr>
<td>2</td>
<td>(1)</td>
<td>(1)</td>
<td>2^{-1} + (2^{-3} - 2^{-1}) \times 2^{-0}</td>
</tr>
<tr>
<td>3</td>
<td>(1, 2)</td>
<td>(1, 2)</td>
<td>2^{-3} + (2^{-2} - 2^{-3}) \times 2^{-1}</td>
</tr>
<tr>
<td>4</td>
<td>(1, 2)</td>
<td>(1, 2, *)</td>
<td>2^{-3} + (2^{-2} - 2^{-3}) \times 2^{-1}</td>
</tr>
<tr>
<td>5</td>
<td>(1, 2, 3)</td>
<td>(1, 2, 3)</td>
<td>2^{-3} + (2^{-2} - 2^{-3}) \times 2^{-1}</td>
</tr>
<tr>
<td>6</td>
<td>(1, 2, 3)</td>
<td>(1, 2, 3, *)</td>
<td>2^{-3} + (2^{-2} - 2^{-3}) \times 2^{-1}</td>
</tr>
<tr>
<td>7</td>
<td>(1, 2, 3)</td>
<td>(1, 2, 3, *, *)</td>
<td>2^{-3} + (2^{-2} - 2^{-3}) \times 2^{-2}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

3 There's no such thing as a free lunch
3.2 Temporal partial metric spaces

▶ Let our cost function \( | \cdot | : X \to (ω \to \mathbb{R}) \) be such that

\[ |X|^n = p_n(x, x) \]

▶ Let \( |¬X|^n = |X|^{|n+1} \) be our way of defining the cost of computing negation. Then we can reason

\[ |¬F|^n = |T|^{|n+1} \]

▶ Suppose now we define (for the purposes of this example) that a formula \( x \) fails if from \( n = 0 \) a cost \( |X|^n \) is computed for some \( n \geq ℵ \) where \( ℵ < ω \) is the maximum cost allowed. Then for our earlier example \( ψ = ¬ψ \) we can reason \( ∀x ≥ ℵ . |ψ|^n \). That is, we can reason monotonically in finite time that the data content of \( ψ \) will not rise above \( |⊥| \), an approximation of its ideal denotation.

3 There's no such thing as a free lunch
3.2 Temporal partial metric spaces

▶ In a demand driven (as opposed to data driven) programming language some potential catastrophes are never encountered. But, for the usual decidability reasons in computation and incompleteness of logic, not all catastrophes can be so avoided.

▶ Our monotonic treatment of Failure takes time may thus be partially correct in the sense of domain theory, but is it so weak as to be useless in practice?

▶ A sequence \( p_0, p_1, \ldots \) of consistent partial metrics as just described is an interesting step forward, but hardly a computable notion of partial metric. That is, is there a notion of partial metric that can express the best and worst of computation?

3 There's no such thing as a free lunch
3.2 Temporal partial metric spaces

▶ Suppose 2 < \( z \). Then \( |¬F|^0 = |¬F|^1 = |T|^2 \) where we totally compute our result before reaching failure. Thus \( |¬T| \) is discernible from \( T \) by means of cost.

▶ The reality of the lack of completeness for any realistic logic of computation implies that any algorithm for defining failure will include the possibility of some cost \( |X|^n \) being necessary to compute \( x \) where \( z < n \). If only \( z \) had been bigger is our feeble lament. This is an example of where the ideal of logical completeness must be substituted by the weaker heuristic of intelligent algorithms.

▶ In contrast to NAF we do not reason non-monotonically to interpret a failure to prove some \( p \) for the logical falsity of \( p \). Our approach is to progress correctly monotonically with partial information as far as is discernibly feasible.

3 There's no such thing as a free lunch
3.2 Temporal partial metric spaces

Although \( ¬⊥ = ⊥ \) in domain theory (as \( ¬F = P \) in logic) we require a model for which the cost of computing \( ¬⊥ \) is discernibly greater than that of computing \( ⊥ \). Consider the following sequence \( p_0, p_1, \ldots \) of partial metrics.

\[ p_0(T, ⊥) = 2^{-n} \]
\[ p_0(F, T) = 2^{-n} + 2^{-1} \]
\[ p_0(F, ⊥) = 2^{-n} + 1 \]
\[ p_0(⊥, ⊥) = 2^{-n} + 1 \]

\( p \) is monotonic in the sense that each \( p_n \) is a partial metric having the usual domain theory ordering \( ⊥ ⊑ F \) and \( ⊥ ⊑ T \), and \( ∀x, y : 0 \leq n < m . p_n(x, y) > p_m(x, y) \)

Note: terms such as \( c + 1 \) presume a suitable predefined algebra/logic of cost.
3 There's no such thing as a free lunch

3.3 Reasonable foundations for cost

Cost is realistic. Our third foundational assumption for cost is that any given mathematical/logical model of computational cost that is wholly a static theory cannot be extended beyond its own limits. In contrast real-world computing is dynamic, showing no signs yet of reaching whatever may prove to be its ultimate limits and impact upon society. Our research continues to thrive upon the assertion that we refute an either all static or all dynamic approach, but rather voluntarily subject ourselves to building a balanced discipline of static & dynamic mathematics/logic for future use.

3.4 Conclusions and further work

▶ Conclusions: metric spaces and topology are excellent for a former age of strong mathematics where cost could be ignored. Now we live in an age where the cost of globalisation is intrinsic to our very survival, calling for compassion back to the world we once ignored. I find the history of metric spaces, topology, logic, incompleteness, computer science, and partial metric spaces to be a fascinating personal journey of discovering inclusion, compassion, & cost in our overly competitive post-Fréchet world.

▶ Further work: there has been an unfortunate cultural divide in Computer Science separating mathematical models & logic from the more successful (recall Nick D’Aloisio) complexity of algorithms. A pronounced winner takes all mentality reminiscent of tragic divides in east meets west Holy Wars pervades today’s IT Wars. The history of Hagia Sophia (Holy Wisdom) from church, to mosque, to secular museum inspires our mathematical research to be a formalism of disparate unity for our world.

“It has long been my personal view that the separation of practical and theoretical work is artificial and injurious. Much of the practical work done in computing, both in software and in hardware design, is unsound and clumsy because the people who do it have not any clear understanding of the fundamental design principles of their work. Most of the abstract mathematical and theoretical work is sterile because it has no point of contact with real computing. One of the central aims of the Programming Research Group as a teaching and research group has been to set up an atmosphere in which this separation cannot happen.”

Christopher Strachey (1916-75)
Founder of the Oxford Programming Research Group (1965)