Real Computable Manifolds and Homotopy Groups

Wesley Calvert

Murray State University

Russell Miller,

Queens College &

Graduate Center – CUNY

September 10, 2009

Unconventional Computation 2009
Universidade dos Açores

Real Computability

Defn.: A *BSS-machine* has an infinite tape, indexed by ω. At each stage, cofinitely many cells are blank, and finitely many contain one number each. In a single step, the machine can copy one cell into another, or perform a field operation ($+$, $-$, $\cdot$, or $\div$) on two cells, or compare any cell to 0 (using $<$ or $=$) and fork, or halt.

The machine starts with a tuple $\vec{p} \in R^{<\omega}$ of parameters in its cells, and the input consists of a tuple $\vec{x} \in R^{<\omega}$, written in the cells immediately following $\vec{p}$. The machine runs according to a finite program, and if it halts within finitely many steps, the output is the tuple of numbers in the cells when it halts.

So BSS programs naturally compute partial functions $R^* \to R^*$, and can be indexed by elements of R^*. When $R = \mathbb{R}$, these are the *real-computable functions*.

Real-Computable Manifolds

A *real-computable d-manifold M* consists of countably many charts U_n, each homeomorphic to $\mathbb{R}^d$ via some α_n, such that $\{U_n\}$ is closed under intersection and covers M. Formally, M is given by real-computable *inclusion functions* i, j, j', and k such that:

- $i(m,n) = 1$ iff $U_m \subseteq U_n$. Then $\varphi_{j(m,n)}$ computes the inclusion map:

  $$\alpha_n^{-1} \circ \varphi_{j(m,n)} \circ \alpha_m : U_m \hookrightarrow U_n$$

  and $\varphi_{j'(m,n)} = \varphi_{j(m,n)}^{-1}$.

- $i(m,n) = 0$ iff $\emptyset \subsetneq U_m \cap U_n \subsetneq U_n$. Then $U_{k(m,n)}$ gives the intersection $U_m \cap U_n$.
- $i(m,n) = -1$ iff $U_m \cap U_n = \emptyset$.

All this is formalized by conditions on i, j, j', and k. The manifold M itself never appears.

It is possible to define connectedness of M just by conditions on i and k.

Paths through M

Classically, a *path* γ is a continuous map: $[0,1] \to M$. In our setting, a path in M is given by $g : [0,1] \to \omega$ and $h : [0,1] \to \mathbb{R}^d$. Then $\gamma(t)$ is the point $h(t)$ in $U_{g(t)}$. We require that there exist $0 = t_0 < \cdots < t_n = 1$ with:

- g constant and h continuous on each $[t_i, t_{i+1})$ and on $[t_{n-1}, 1]$.
- The point $h(t_i)$ in $U_{g(t_i)}$ is the limit of $h(t)$ in $U_{g(t)}$ as $t \to t_i^-$, for each $i < n$.

This path is real-computable if g and h are. n and $\langle t_0, \ldots, t_n \rangle$ are finitely much info, but are not normally assumed to be given.

A path is a *loop* if $g(0) = g(1)$ and $h(0) = h(1)$ (so $\gamma(0) = \gamma(1)$).

Paths and Homotopy

Fact: Every path in M is homotopic to a computable path: the straight lines in each $U_{g(t_i)}$ from $h(t_i)$ to $\lim_{t \to t_{i+1}^-} h(t)$. Indeed, the path is determined up to homotopy by the sequence

$$\langle g(t_0), g(t_1), \ldots, g(t_n) \rangle \in \omega^{<\omega}.$$

An index for a homotopic computable path can be real-computed from n and an index for g, provided $g(t_i) \neq g(t_{i+1})$ for all $i < n$.

This relies on our convention that all charts U_m are connected. Classically, this convention is easy; but how effective is the classical argument?

Noncomputable Nullhomotopy

However, indices for g and h (without n) are insufficient to decide whether the path is nullhomotopic, unless M is simply connected.

Proof: For any real-computable ψ, let $f = \langle g, h \rangle$ be a loop not homotopic to a constant. Define the computable loop $\varphi_{\vec{e}}$:

$$\varphi_{\vec{e}}(t) = \begin{cases} f(2 + (t-1)2^{s+1}), & \text{if } \psi_{=s}(\vec{e})\!\downarrow = 1 \ \& \\ & t \in [\frac{2^s - 1}{2^s}, \frac{2^{s+1}-1}{2^{s+1}}) \\ \text{base point of } f, & \text{if not, or if } t = 1. \end{cases}$$

So $\varphi_{\vec{e}}$ goes around f once iff ψ says $\varphi_{\vec{e}}$ is nullhomotopic.

This uses the **Recursion Theorem** for real computability.

Simple Connectedness

Likewise, no real-computable ψ can decide whether the manifold given by (indices for) i, j, j', and k is simply connected. We can define these functions to produce six charts forming an M with $\pi_1(M) \cong \mathbb{Z}$. Then, if ψ ever halts on those indices, they put a new chart in the middle, making M simply connected.

Thus the property of not being simply connected is not even real-semidecidable (= domain of a real-computable function). Nor is simple connectedness real-semidecidable, by a similar argument. Indeed, these properties are not decidable by any function of the form $\lim_s \theta(i, j, j', k, s)$ with θ real-computable.

Computing $\pi_1(M)$

Modulo homotopy, a path $\langle g, h\rangle$ can be described completely by the sequence $\langle g(t_0), g(t_1), \ldots, g(t_n)\rangle$. So we think of elements $\vec{x} \in \omega^{<\omega}$ as (possible) loops. To be a loop, $\vec{x}$ must have $x_0 = x_n$ (= some particular m, where U_m contains our base point) and $i(x_p, x_{p+1}) \neq -1$ for all $p < n$. All this is real-computable.

In real-computability, a single real may now be given which codes all necessary information for determining a set of homotopy class representatives and computing $\pi_1(M)$. In truth, this is now a problem for Turing machines, not BSS-machines.

Computing Homotopy

Lemma: Homotopy of two paths α and β from x to y in M is computably enumerable, relative to the inclusion functions i and k.

Proof: By compactness, the image of the set of all paths used in a homotopy must be contained in finitely many charts. The proof is then just an induction, using:

Fact: For α and β as shown, $\alpha \simeq \beta$ within $U_m \cup U_n \cup U_p$ iff $U_m \cap U_n \cap U_p \neq \emptyset$.

Cf. work of Brown, *Annals of Math. 1957.*

Presentation of $\pi_1(M)$

The inclusion functions i and k are functions on ω, and j and j' are not needed. Relative to i and k, $\pi_1(M)$ is presentable as the quotient of a Turing-computable subset of $\omega^{<\omega}$ by the c.e. equivalence relation of homotopy.

Likewise, the operation of concatenation is Turing-computable on this set.

The question remains: how difficult is it to start with a more general real-computable presentation of a manifold and produce a presentation which has all charts connected and is closed under intersection of charts? And is this an appropriate question for real computability, or for Turing computability?