Real Computable Manifolds

Wesley Calvert

Murray State University

Russell Miller,

Queens College &

Graduate Center – CUNY

January 8, 2009

AMS Special Session on
Orderings in Logic and Topology

AMS-MAA 2009 Joint Meetings
Washington, DC

Computability on $\mathbb{N}$

Turing computability: an idealized computer accepts finite binary strings (or finite tuples from $\mathbb{N}$) as inputs, runs according to a finite program, and may halt within finitely many steps, outputting another binary string or tuple from $\mathbb{N}$.

So Turing programs naturally compute partial functions $\mathbb{N}^j \to \mathbb{N}^k$ or $\mathbb{N}^* \to \mathbb{N}^*$. (*Partial*: the domain may be a proper subset of $\mathbb{N}^j$ or $\mathbb{N}^*$.)

Halting Problem: does a given Turing program with a given input ever halt? No Turing machine can give you the correct answer in all cases.

A subset of $\mathbb{N}^*$ is *computable* iff its characteristic function is computable.

Computability on $\mathbb{R}$

Blum-Shub-Smale computability (or **real computability):** a *BSS machine* accepts finite tuples from $\mathbb{R}$ as inputs, runs according to a finite program, which has finitely many reals as parameters and can perform operations and comparisons on reals. It may halt within finitely many steps, outputting another tuple from $\mathbb{R}$.

So BSS programs naturally compute partial functions $\mathbb{R}^* \to \mathbb{R}^*$, and can be indexed by elements of $\mathbb{R}^*$.

Halting Problem: does a given BSS program with a given input ever halt? Again, no BSS machine can give you the correct answer in all cases.

Real Computable Manifolds

Defn.: A *real-computable n-manifold M* consists of (1) a computable subset $C \subseteq \mathbb{R}^*$; and (2) real-computable i, j, k, the *inclusion functions*, satisfying the conditions on the next slide.

Interpretation:

- Each $\vec{r} \in C$ is a chart $U_{\vec{r}}$ in M, with domain $\mathbb{R}^n$;
- $i(\vec{q}, \vec{r}) = 1$ iff $U_{\vec{q}} \subseteq U_{\vec{r}}$, and then $j(\vec{q}, \vec{r})$ is an index for the (computable!) inclusion map;
- If $i(\vec{q}, \vec{r}) = 0$, then $k(\vec{q}, \vec{r}) \in C^*$ and $\sqcup_{\vec{t} \in k(\vec{q}, \vec{r})} U_{\vec{t}} = U_{\vec{q}} \cap U_{\vec{r}}$.
- Else $i(\vec{q}, \vec{r}) = -1$, and $U_{\vec{q}} \cap U_{\vec{r}} = \emptyset$.

Conditions on C, i, j, and k

If $i(\vec{t},\vec{q}) = i(\vec{q},\vec{r}) = 1$, then $i(\vec{t},\vec{r}) = 1$ and

$$\varphi_{j(\vec{q},\vec{r})} \circ \varphi_{j(\vec{t},\vec{q})} = \varphi_{j(\vec{t},\vec{r})}.$$

Also, $(\forall \vec{q},\vec{r} \in C)$ i on input $(\vec{q},\vec{r})$ outputs either

- 1, and $\varphi_{j(\vec{q},\vec{r})}$ is a total real-computable homeomorphism from $\mathbb{R}^n$ into $\mathbb{R}^n$. ($\varphi_{j(\vec{q},\vec{r})}$ then describes the inclusion $U_{\vec{q}} \subseteq U_{\vec{r}}$.)
- 0, and $k(\vec{q},\vec{r}) = \vec{t}$ s.t. $i(\vec{t},\vec{q}) = i(\vec{t},\vec{r}) = 1$ & $\forall \vec{u},\vec{v} \in C[i(\vec{u},\vec{q}) = i(\vec{u},\vec{r}) = 1 \implies i(\vec{u},\vec{t}) = 1]$ & if $i(\vec{q},\vec{v}) = i(\vec{r},\vec{v}) = 1$, then $\text{range}(\varphi_{j(\vec{r},\vec{v})}) \cap \text{range}(\varphi_{j(\vec{q},\vec{v})}) = \text{range}(\varphi_{j(\vec{t},\vec{v})})$. (Here $U_{\vec{t}} = U_{\vec{q}} \cap U_{\vec{r}}$.)
- -1, and $(\forall \vec{u},\vec{v} \in C)[i(\vec{u},\vec{q}) \neq 1$ or $i(\vec{u},\vec{r}) \neq 1]$ & if $i(\vec{q},\vec{v}) = i(\vec{r},\vec{v}) = 1$, then $\text{range}(\varphi_{j(\vec{q},\vec{v})}) \cap \text{range}(\varphi_{j(\vec{r},\vec{v})}) = \emptyset$. (Here $U_{\vec{q}} \cap U_{\vec{r}} = \emptyset$.)

Loops and Homotopy

Defn.: A *loop* in M is given by finitely many continuous functions $f_m : [t_{m-1}, t_m] \to \mathbb{R}^n$, where $0 = t_0 < \cdots < t_l = 1$, along with $\vec{r}_1, \ldots, \vec{r}_l \in C$. We think of f mapping $[0, 1]$ into M by mapping each $[t_{m-1}, t_m]$ into $U_{\vec{r}_m}$, with the obvious condition on the end points. If all f_m are computable, then the loop is computable.

Fact: Every loop in M is homotopic to a computable loop.

(One could define *computable homotopy*, but for now we just use homotopy.)

Noncomputable Nullhomotopy

Build a computable 2-manifold M with charts indexed by $\mathbb{N} \times \mathbb{R}^*$:

- $U_{0,\vec{r}}$ and $U_{1,\vec{r}}$ form an annulus.
- Define a computable loop $f_{\vec{r}}$ around this annulus.
- For $s > 1$, if $\varphi_{\vec{r}}(f_{\vec{r}})$ halts in exactly $(s-1)$ steps and says that $f_{\vec{r}}$ is *not* nullhomotopic, then $U_{s,\vec{r}}$ fills in the hole in the annulus.
- If no halt occurs at step $(s-1)$, then $U_{s,\vec{r}}$ is disjoint from all other charts.

So no $\varphi_{\vec{r}}$ correctly decides nullhomotopy of $f_{\vec{r}}$.

A simpler manifold

The above M has no countable cover. But even in S^1, there is no real-computable ψ which accepts $\vec{r}$ as input and satisfies:
if $\varphi_{\vec{r}}$ is a loop in S^1, then

$$\psi(\vec{r}) = \begin{cases} 1, & \text{if } \varphi_{\vec{r}} \text{ nullhomotopic} \\ 0, & \text{if not.} \end{cases}$$

Proof: Use the Recursion Thm. for BSS-machines to produce $\varphi_{\vec{r}} : [0,1] \to S^1$ s.t. $\varphi_{\vec{r}}(0) = \varphi_{\vec{r}}(1) = 1$ and

$$\varphi_{\vec{r}} \restriction \left[\frac{1}{2^s}, \frac{1}{2^{s+1}}\right] = \begin{cases} S^1, & \text{if } \psi(\vec{r}) = 1 \text{ in} \\ & \text{exactly } s \text{ steps} \\ 1, & \text{if not.} \end{cases}$$

General Theorems

The procedure above works for any computable M containing a computable loop which is not nullhomotopic.

Thm. (Calvert-M.): For any real-computable manifold M, TFAE:

1. There exists a real-computable ψ such that ($\forall$ computable loops $\varphi_{\vec{r}}$ in M) $\psi(\vec{r})$ decides nullhomotopy of $\varphi_{\vec{r}}$,
2. All computable loops in M are nullhomotopic.
3. M is simply connected.

Thm. (Calvert-M.): Simple-connectedness is not decidable. That is, there is no real-computable ψ such that whenever $\vec{r}$ is the index of a computable manifold M, $\psi(\vec{r})$ decides whether M is simply connected.