Computing the Fundamental Group of an $\mathbb{R}$-computable Manifold *

Wesley Calvert[†] & Russell Miller[‡]

September 27, 2009

Abstract

Using the model of $\mathbb{R}$-computability developed by Blum, Cucker, Shub, and Smale, we investigate the difficulty of determining the answers to several basic topological questions about manifolds. Under natural definitions of $\mathbb{R}$-computable manifold and $\mathbb{R}$-computable path, we show that, while BSS machines cannot in general decide such questions as nullhomotopy and simple connectedness for such structures, there are nevertheless $\mathbb{R}$-computable presentations of paths and homotopy equivalence classes under which such computations are possible. Indeed, we argue that the appropriate model of computation for homotopy questions is the Turing model, not the BSS model for $\mathbb{R}$.

1 Introduction

A notable shortcoming of the standard (Turing) model of computation is that it does not produce a theory of effectiveness relevant to uncountable structures. Since these structures are a routine part of the practice of pure and applied mathematics, a growing body of literature has addressed effective

*A significant portion of this article appeared as [5], in the proceedings volume of the conference *Unconventional Computation 2009*.

†Both authors were partially supported by Grant #13397 from the Templeton Foundation, and wish to acknowledge useful conversations with Kerry Ojakian.

‡The second author was partially supported by grants numbered 61467-00 39 and 62632-00 40 from The City University of New York PSC-CUNY Research Award Program.

mathematics on uncountable structures (see, for instance [1, 4, 6, 7, 8, 10, 11, 12]). The present paper will begin to describe the use of one of the proposed models to explore the effective homotopy theory of manifolds.

In [2], Blum, Shub, and Smale introduced a notion of computation based on full-precision real arithmetic, which received its canonical form in [1]. Let $\mathbb{R}^\infty$ be the set of finite sequences of elements from $\mathbb{R}$, and $\mathbb{R}_\infty$ the bi-infinite direct sum

$$\bigoplus_{i\in\mathbb{Z}} \mathbb{R}.$$

Definition 1.1. A machine M over $\mathbb{R}$ is a finite connected directed graph, containing five types of nodes: input, computation, branch, shift, and output, with the following properties:

1. The unique input node has no incoming edges and only one outgoing edge.
2. Each computation and shift node has exactly one output edge and possibly several input branches.
3. Each output node has no output edges and possibly several input edges.
4. Each branch node η has exactly two output edges (labeled 0_η and 1_η) and possibly several input edges.
5. Associated with the input node is a linear map $g_I : \mathbb{R}^\infty \to \mathbb{R}_\infty$.
6. Associated with each computation node η is a rational function $g_\eta : \mathbb{R}_\infty \to \mathbb{R}_\infty$.
7. Associated with each branch node η is a polynomial function $h_\eta : \mathbb{R}_\infty \to \mathbb{R}$. (The output edges 0_η and 1_η are associated to the conditions $h_\eta \geq 0$ and $h_\eta < 0$, respectively.)
8. Associated with each shift node is a map $\sigma_\eta \in \{\sigma_l, \sigma_r\}$, where $\sigma_l(x)_i = x_{i+1}$ and $\sigma_r(x)_i = x_{i-1}$.
9. Associated with each output node η is a linear map $O_\eta : \mathbb{R}_\infty \to \mathbb{R}^\infty$.

Each machine computes a partial function from $\mathbb{R}^\infty$ to $\mathbb{R}^\infty$ in the natural way. Such a function is said to be $\mathbb{R}$-computable. Our notation mirrors that of Turing computability, in that we index $\mathbb{R}$-computable functions (effectively) as $\varphi_{\vec{e}}$, where $\vec{e}$ is allowed to range over $\mathbb{R}^\infty$. Of course, the programs of such functions are finite, but each program is allowed to use finitely many real parameters, yielding 2^ω-many such programs. This precludes procedures which simultaneously run all $\mathbb{R}$-computable functions on an input (let alone on all possible inputs), but diagonalization arguments are often still possible. We have an $\mathbb{R}$-computability version of Kleene's Recursion Theorem.

Theorem 1.2 (Recursion Theorem)**.** *For every $\mathbb{R}$-computable function f with domain $\mathbb{R}^\infty$, there exists an $\vec{x} \in \mathbb{R}^\infty$ such that $\varphi_{\vec{x}} = \varphi_{f(\vec{x})}$.*

Both this theorem and its proof are very similar to Kleene's original theorem for Turing computability; see for example [13, II.3.1]

In the present paper, we will explore manifolds which are given effectively (in the sense of $\mathbb{R}$-computation). In the remainder of this section, we describe this sense of effectiveness exactly, and will recall some relevant definitions from topology. In Section 2, we show that our definition of $\mathbb{R}$-computable manifold, despite being quite natural and apparently using $\mathbb{R}$-computability in an essential way, largely reduces to its Turing-computable aspects. In Section 3, we will describe some ineffectiveness results in homotopy theory: that nullhomotopy and simple-connectedness are both $\mathbb{R}$-undecidable. In Section 4, we will show that certain important standard computations — notably the computation of the fundamental group — are still, modulo the difficulties in Section 3, $\mathbb{R}$-computable. Indeed, we will argue there that $\mathbb{R}$-computability is an overly strong notion of computability for these topological considerations, and that the more appropriate setting for such a discussion is that of traditional Turing computability. (Of course, Turing computability can be viewed as computability in the BCSS model, only over the ring $\mathbb{Z}/(2)$ instead of the field $\mathbb{R}$, as discussed in [1].) Section 5 then suggests further avenues of investigation where $\mathbb{R}$-computability might still be essential.

The definitions in this section and the noncomputability results in Section 3 have appeared previously in the extended abstract [5] from the proceedings volume of the conference Unconventional Computation 2009. We include them here because that volume may not have been widely distributed. The thrust of this paper is distinct from [5]: our reduction from the natural definition of $\mathbb{R}$-computable manifold to the Turing model of computation, which is based on Sections 2 and 4. We believe that the results there are original in this context, though their outcomes may be seen as related to results on manifolds known before the development of the BSS model.

Definition 1.3. An $\mathbb{R}$-*computable manifold* M, of *dimension* d, consists of four *inclusion functions* i, j, j', and k, all $\mathbb{R}$-computable partial functions on the domain ω^2, satisfying the following conditions for all $m, n \in \omega$. (Interpretation of these conditions appears below.)

1. If $i(m,n)\downarrow= 1$, then $\varphi_{j(m,n)}$ is a total $\mathbb{R}$-computable homeomorphism from $\mathbb{R}^d$ into $\mathbb{R}^d$, and $\varphi_{j'(m,n)} = \varphi^{-1}_{j(m,n)}$, and $k(m,n)\downarrow= k(n,m)\downarrow= m$.

2. If $i(m,n)\downarrow= 0$, then all of the following hold:

   - $k(m,n)\downarrow= k(n,m)\downarrow\in \omega$ with $i(k(m,n),m) = i(k(m,n),n) = 1$; and
   - for all $p \in \omega$, if $i(p,m) = i(p,n) = 1$, then $i(p,k(m,n)) = 1$; and
   - for all $q \in \omega$, if $i(m,q) = i(n,q) = 1$, then $i(k(m,n),q) = 1$ with

   $$\text{range}(\varphi_{j(m,q)}) \cap \text{range}(\varphi_{j(n,q)}) = \text{range}(\varphi_{j(k(m,n),q)}).$$

3. If $i(m,n) \notin \{0,1\}$, then $i(m,n)\downarrow= i(n,m)\downarrow= -1$, and

   $$(\forall p \in \omega)[i(p,m) \neq 1 \text{ or } i(p,n) \neq 1],$$

   and for all $q \in \omega$, if $i(m,q)$ and $i(n,q)$ both lie in $\{0,1\}$, then

   $$\text{range}(\varphi_{j(k(m,q),q)}) \cap \text{range}(\varphi_{j(k(n,q),q)}) = \emptyset.$$

4. For all $q \in \omega$, if $i(m,n) = i(n,q) = 1$, then $i(m,q) = 1$ and

   $$\varphi_{j(n,q)} \circ \varphi_{j(m,n)} = \varphi_{j(m,q)}.$$

5. $i(m,m)\downarrow= 1$.

So i and k are actually functions from ω^2 to ω (total and partial, respectively). j and j' also accept only inputs from ω^2, but output elements of $\mathbb{R}^\infty$, namely indices for $\mathbb{R}$-computable functions. (Remember that such indices are themselves given by real numbers.) Of course, every function from $\omega^{<\omega}$ to itself is $\mathbb{R}$-computable, since a single real parameter can hold all information about the function and the program can break down that parameter and recover any finite amount of this information. Therefore, the requirement by Definition 1.3 that i and k be $\mathbb{R}$-computable is actually a dead letter;

$\mathbb{R}$-computability of j and j', along with satisfaction of the five conditions, is the essence of the definition.

Now we explain how this very abstract definition is to be understood. First, each natural number m represents a chart U_m for the manifold M, which is to say, a nonempty open subset of M homeomorphic to $\mathbb{R}^d$ via some map

$$\alpha_m : \mathbb{R}^d \to U_m.$$

In fact, though, we do not give this homeomorphism, since we do not wish to attempt to present the points in the manifold globally in an effective way. Instead we simply understand $\mathbb{R}^d$ to represent the chart U_m, and understand the manifold M to be the union of this countable collection of charts. The meat of the definition lies in the inclusion functions, which describe the inclusion relations among the charts. For each m and n, $i(m,n)$ equals 1, 0, or -1, according as either $U_m \subseteq U_n$, or $\emptyset \subsetneq U_m \cap U_n \subsetneq U_m$, or $U_m \cap U_n = \emptyset$, and the conditions translate as follows.

First, if $i(m,n) = 1$, then $j(m,n)$ is the index for an $\mathbb{R}$-computable map from $\mathbb{R}^d$ into $\mathbb{R}^d$ which describes the inclusion map $U_m \hookrightarrow U_n$, in the sense that $\alpha_n \circ \varphi_{j(m,n)} \circ \alpha_m^{-1}$ is this inclusion map. Moreover, we can also compute an index $j'(m,n)$ for its inverse. Condition 5 specifies that this holds when $m = n$, and one proves from Condition 4 that $\varphi_{j(m,m)}$ is the identity map.

If $i(m,n) = 0$, then $U_m \not\subseteq U_n$, but the two charts have nonempty intersection $U_{k(m,n)}$. (Notice that this implies that our collection of charts must be closed under finite intersection, and that the intersection of two charts must be homeomorphic to $\mathbb{R}^d$.) The conditions ensure that $U_{k(m,n)}$ contains every other chart U_p which is a subset of both U_m and U_n, and also that if a chart U_q contains both U_m and U_n, then it must contain $U_{k(m,n)}$, which must be the intersection of U_m with U_n within U_q.

Finally, if $i(m,n) = -1$, then the conditions ensure that $U_m \cap U_n$ does not contain any other U_p, and that for any U_q intersecting both U_m and U_n, we have $U_m \cap U_n = \emptyset$ within U_q. (There is no harm in assuming that $k(m,n)\uparrow$ in this case.)

Condition 4 is the obvious rule for composition of inclusion maps: if $U_m \subseteq U_n \subseteq U_q$, then

$$\alpha_q \circ \varphi_{j(m,q)} \circ \alpha_m^{-1} = (\alpha_q \circ \varphi_{j(n,q)} \circ \alpha_n^{-1}) \circ (\alpha_n \circ \varphi_{j(m,n)} \circ \alpha_m^{-1}).$$

As a quick example, we prove here that the intersection operation is indeed associative. The lemma expresses the idea that $(U_m \cap U_n) \cap U_p$ should equal $U_m \cap (U_n \cap U_p)$ whenever the former is nonempty.

Lemma 1.4. *If $i(m,n) \neq -1$ and $i(k(m,n),p) \neq -1$, then $k(k(m,n),p) = k(m,k(n,p))$.*

Proof. Parentheses denote the conditions of Definition 1.3 used in the argument. We know by (1) and (2) that $i(k(k(m,n),p),k(m,n)) = 1$, and by (2) that $i(k(m,n),m) = i(k(m,n),n) = 1$, so (4) forces $i(k(k(m,n),p),m) = i(k(k(m,n),p),n) = 1$. Moreover $i(k(k(m,n),p),p) = 1$, according to (2), which also then shows $i(k(k(m,n),p),k(n,p)) = 1$. One last use of (2) yields $i(k(k(m,n),p),k(m,k(n,p))) = 1$.

Likewise $i(k(m,k(n,p)),k(m,n)) = i(k(m,k(n,p)),p) = 1$, and it follows that $i(k(m,k(n,p)),k(k(m,n),p)) = 1$. Hence

$$\begin{aligned} k(m,k(n,p)) &= k(k(m,k(n,p)),k(k(m,n),p)) \\ &= k(k(k(m,n),p),k(m,k(n,p))) \\ &= k(k(m,n),p), \end{aligned}$$

with the first equality following from this paragraph and the last from the preceding paragraph (both using (1)), and the middle equality from the symmetry of k. □

An earlier version of this definition indexed the charts by finite tuples $\vec{x} \in \mathbb{R}^\infty$ of real numbers, with the condition that the indices must be precisely the tuples from some fixed $\mathbb{R}$-computable subset of $\mathbb{R}^\infty$. This allowed the possibility that the charts form an uncountable collection (but still of cardinality $\leq 2^\omega$), and also allowed those charts to be enumerated in a far less effective way. Since most definitions of manifold require the manifold to be covered by a countable collection of charts, we feel that Definition 1.3 reflects the usual topological situation more accurately. (Since the property of being covered by countably many charts is equivalent to that of having a countable basis, manifolds having these properties are often called *second-countable* manifolds in the literature.) It also gives a more precise enumeration of the charts, allowing us greater ability to search through the charts to find the one we need, and this permits us to prove stronger theorems about our $\mathbb{R}$-computable manifolds, at the cost of excluding those which are not second-countable or whose presentations are insufficiently effective to fit our definition. This paper is intended in part to show that the choice of the stronger definition makes Turing computation, rather then $\mathbb{R}$-computation, the natural setting. By inference, then, the Turing model is the appropriate model of computation for all topological operations on second-countable manifolds.

Our definition of manifold is also strict in requiring that the intersection of any two charts must be another chart (or else empty), and hence must be connected, indeed simply connected. For manifolds in general this is not usually required, but on the other hand, it is usually possible to take a manifold in which the intersection of two charts need not be simply connected and to produce a new, finer cover of it by an atlas of simply connected charts closed under intersections. We leave open the question of how effectively such a transformation of the cover can be accomplished. Our definition does facilitate the enumerability (and $\mathbb{R}$-decidability) of homotopy for certain specific classes of computable paths, a question which we conjecture is not enumerable in the more general context. This conjecture would imply that a general cover cannot always be effectively converted into a cover with pairwise intersections all simply connected.

Clearly Definition 1.3 is quite abstract, in the sense that it essentially ignores the intended underlying topological space M entirely, giving a set of conditions on the charts (each of which is a copy of $\mathbb{R}^d$), rather than the corresponding conditions on the space M itself. We refer the reader to [15, p. 30, Example 8] for a noneffective precursor to this definition. That example also provides a condition on the charts which is equivalent to connectedness of M:

Definition 1.5. The $\mathbb{R}$-computable manifold M is *connected* if there is no proper nonempty subset S of ω such that

- whenever $i(m,n) = 0$ with $m,n \in S$, we also have $k(m,n) \in S$; and
- whenever $i(m,n) = 0$ with $m,n \notin S$, we also have $k(m,n) \notin S$; and
- whenever $i(m,n) = 1$ with $m \in S$, we also have $n \in S$; and
- whenever $i(m,n) = 1$ with $m \notin S$, we also have $n \notin S$.

If M is not connected, then we define its connected components to be the maximal subsets of ω satisfying all four of the above properties. (Intuitively, the connected components of M are the unions of the charts corresponding to these maximal subsets.)

Classically, a *path* in a manifold M is a continuous map of the closed unit interval into M. Two paths p_0 and p_1 with the same endpoints ($p_0(0) = p_1(0)$ and $p_0(1) = p_1(1)$) are said to be *homotopic* if there is a continuous map $h : I \times I \to M$ such that $h(t, 0) = p_0(t)$ and $h(t, 1) = p_1(t)$ and $h(0, s) = p_1(0)$ and $h(1, s) = p_1(1)$ for all s, t. A path is said to be a *loop* if and only if its endpoints coincide. The class of loops in a connected manifold M up to homotopy equivalence forms a group under concatenation, called the *fundamental group* of M, or $\pi_1(M)$. A loop in the identity class is said to be *nullhomotopic*. A manifold M such that $\pi_1(M)$ is trivial is said to be simply connected. More detailed background on these issues can be found in [9]. With our abstract definition, we are forced into a more involved definition of path.

Definition 1.6. A *path* in a $\mathbb{R}$-computable manifold M, with inclusion functions i, j, and k, consists of a pair of functions $g : [0, 1] \to \omega$ and $h : [0, 1] \to \mathbb{R}^d$ for which there exists a finite sequence $0 = t_0 < t_1 < \cdots < t_n = 1$ of real numbers such that, for all $m < n$:

- $g{\restriction}[t_m, t_{m+1})$ is constant, with $g(1) = g(t_{n-1})$; and
- $h{\restriction}[t_m, t_{m+1})$ is continuous (and right-continuous at t_m), and h is also left-continuous at 1; and
- $i(g(t_m), g(t_{m+1})) \in \{0, 1\}$ and

$$\lim_{x \to t_{m+1}^-} \varphi_{j(k(g(t_m),g(t_{m+1})),g(t_{m+1}))}(\varphi^{-1}_{j(k(g(t_m),g(t_{m+1})),g(t_m))}(h(x))) = h(t_{m+1}).$$

If g and h are both $\mathbb{R}$-computable functions, then we call f a *computable path*. If $g(0) = g(1)$ and $h(0) = h(1)$, then the path is a *loop*.

The intuition here is that the t_m are values at which the path switches from one chart to another. Of course, they are not uniquely defined, but by compactness of $[0, 1]$, we need only finitely many such points to express the entire path. We do *not* require any such sequence of points to be computable. Of course, any finite sequence of real numbers is immediately $\mathbb{R}$-computable, but for an infinite collection of paths $p_{\vec{c}}$, indexed by $\vec{c}$ in some $\mathbb{R}$-computable set C, we will call the collection *computable* if there is a single $\mathbb{R}$-computable p satisfying $p(\vec{c}, x) = p_{\vec{c}}(x)$, and *strongly computable* if an appropriate n and $t_1 < \cdots < t_{n-1}$ for each $p_{\vec{c}}$ can be computed uniformly in $\vec{c}$.

Weyl himself accepted the difficulty of carrying out homotopy theory on manifolds defined in this way as a drawback of this definition. He noted that Weil's criticism [14] of the introduction of homology via the Eilenberg-Steenrod axioms applied also to this definition of a manifold. In essence, Sections 3 and 4 of the present paper argue the two sides of this criticism. Weil claims that "triangulation is a quite trivial matter," and gives the structure of the fundamental group. This may be contrasted with a more abstract presentation, like the definition above, in which the computation of the fundamental group — in the views of both Weil and Weyl — is presumably more difficult. One understanding of Section 3 is that triangulation — at least, when starting with a manifold in the way we have defined it — is quite difficult. On the other hand, one understanding of Section 4 is that once one can do that, a presentation of the fundamental group is indeed possible.

2 Reducing to Computation on ω

To show that Definition 1.3 is equivalent to the more intuitive definition of a topological manifold (as an actual topological space with an atlas of charts), one starts with the inclusion functions from Definition 1.3 and produces a topological manifold for which ω-many copies of $\mathbb{R}^d$ form a cover via inclusion functions $\alpha_m : \mathbb{R}^d \to U_m$, as in the discussion in the previous section. The domain of this manifold is $(\mathbb{R}^d) \times \omega$, modulo the equivalence relation equating $(\vec{x}, m)$ with $(\vec{y}, n)$ whenever there exist $\vec{z}$ and p with $i(p, m) = i(p, n) = 1$ and $\varphi_{j(p,m)}(\vec{z}) = \vec{x}$ and $\varphi_{j(p,n)}(\vec{z}) = \vec{y}$. The chart U_m is the set of classes containing points from $\mathbb{R}^d \times \{m\}$, and $\alpha_m(x)$ is the class of (x, m). The inclusion functions are then shown to satisfy the intuitive roles described for them, and one checks that this is a topological manifold, unique up to homeomorphism. All this is standard topology, going back to the introduction of (the noneffective version of) Definition 1.3 as an alternative to the topological version.

When we follow the same process with considerations of effectiveness in mind, a similar argument applies. The result of the argument, however, is that the notion of $\mathbb{R}$-computability actually drops out of the picture, leaving only issues of computability on the natural numbers. The conclusion of this section will be that in fact, the appropriate notion of effectiveness for topological manifolds is Turing computability. In Sections 3 and 4 we extend this argument to homotopy questions about paths through $\mathbb{R}$-computable manifolds, while in Section 5 we address situations where $\mathbb{R}$-computability may still be required.

The key to our results is the following proposition.

Proposition 2.1. *Let i_0, j_0, j_0', k_0, i_1, j_1, j_1', and k_1 be $\mathbb{R}$-computable functions defining two $\mathbb{R}$-computable manifolds M_0 and M_1. If $i_0 = i_1$ and $k_0 = k_1$, then M_0 and M_1 are homeomorphic.*

Proof. Of course, here we are thinking of homeomorphism between actual topological manifolds corresponding to M_0 and M_1. These could be the ones defined above, but we simply assume that we have functions $\alpha_m : \mathbb{R}^d \to M_0$ defining an atlas of charts $U_m = \text{range}(\alpha_m)$ on M_0, and corresponding $\beta_m : \mathbb{R}^d \to M_1$ defining charts V_m on M_1. (So the notion of homeomorphism here is the classical one, not any abstract version cooked up to match Definition 1.3.)

For any $x \in M_0$, we know that there exists an m with $x \in U_m^0$. We define $f(x) = \beta_m(\alpha_m^{-1}(x))$. It is necessary to show that this f is well-defined, and then that f is a homeomorphism from M_0 onto M_1.

For well-definedness, suppose that $x \in U_m \cap U_n$, so that either m or n could be used to define $f(x)$. Then $i(m,n)$ is either 1 or 0. Suppose $i(m,n) = 1$. Then $\alpha_n \circ \varphi_{j(m,n)} \circ \alpha_m^{-1}(x) = x$, since this is just the inclusion map of U_m into U_n, and likewise $\beta_n \circ \varphi_{j(m,n)} \circ \beta_m^{-1}(f(x)) = f(x)$. So:

$$\begin{aligned}
\beta_n(\alpha_n^{-1}(x)) &= \beta_n(\alpha_n^{-1}(\alpha_n \circ \varphi_{j(m,n)} \circ \alpha_m^{-1}(x))) \\
&= \beta_n(\varphi_{j(m,n)} \circ (\beta_m^{-1} \circ \beta_m) \circ \alpha_m^{-1}(x))) \\
&= (\beta_n \circ \varphi_{j(m,n)} \circ \beta_m^{-1})(f(x)) \\
&= f(x).
\end{aligned}$$

In case $i(m,n) = 0$, similar arguments show that $\beta_{k(m,n)}(\alpha_{k(m,n)}^{-1}(x)) = f(x)$, and also that $\beta_{k(m,n)}(\alpha_{k(m,n)}^{-1}(x)) = \beta_n(\alpha_n^{-1}(x))$. This completes the proof that f is well-defined.

Our map f is bijective, simply because it has a well-defined inverse given by $g(y) = \alpha_m(\beta_m^{-1}(y))$, where $y \in V_m$; the proof is exactly the same as for f. And since the maps α_m and β_m are assumed to be homeomorphisms, we see that $f \restriction U_m$ is continuous for every m, and likewise $g \restriction V_m$. Because continuity is a local property, this shows that f and g are both continuous, hence are themselves homeomorphisms. □

Thus, an $\mathbb{R}$-computable manifold M may be specified up to homeomorphism just by the inclusion functions i and k, which are countable: the total function i maps ω^2 into $\{1, 0, -1\}$, and the partial function k maps ω^2 into ω. In fact, every second-countable topological manifold with an atlas of simply-connected charts, closed under intersection, has inclusion functions i, j, j', and k as required by Definition 1.3, except that the inclusion maps $\varphi_{j(m,n)}$ and their inverses may not be $\mathbb{R}$-computable. (Just fix a collection of maps $\alpha_m : \mathbb{R}^d \to U_m$ giving the charts, and let $\varphi_{j(m,n)}$ be the function $(\alpha_n^{-1} \circ \alpha_m)$ whenever $i(m, n) = 1$.) So, even when such a manifold fails to be homeomorphic to an $\mathbb{R}$-computable manifold, we can still consider it, up to homeomorphism, using its functions i and k. This is discussed a bit further in Section 5.

Of course, the countability of i and k rests on the assumption of second countability of M, but most topologists assume second countability even as part of the definition of manifold. (And without any cardinality assumption, there is no reason to expect $\mathbb{R}$-computability to apply, since M could have a base of cardinality $> 2^\omega$.) In any event, the functions j and j', which are the only two functions in Definition 1.3 that actually involve inputs or outputs from $\mathbb{R}$, are irrelevant, at least up to classical homeomorphism. This makes it much more natural to consider Turing computability of the functions i and k, and to define the degree of M to be the join of the Turing degrees of i and k, rather than remaining in the framework of $\mathbb{R}$-computation. However, we shall wait and consider specific paths through the manifold, still under $\mathbb{R}$-computability, before returning in Section 4 to Turing computability.

3 The Undecidability of Nullhomotopy and Simple Connectedness

Having seen that our very natural definition of $\mathbb{R}$-computable manifold turns out to reduce (up to classical homeomorphism) to questions about computability on ω, we ask whether questions about individual paths do the same. Of course, two homeomorphic manifolds must have the same fundamental group, but it is quite plausible for questions about specific paths to involve the functions j and j' of an $\mathbb{R}$-computable manifold, and thus to demand consideration within the framework of $\mathbb{R}$-computability. We will show in this section that this is partially true: j and j' cannot be disregarded, but on the other hand, even $\mathbb{R}$-computability is not powerful enough to decide questions such as the nullhomotopy of loops.

Although this section is dedicated to proofs of undecidability, it will be useful for us to begin with a positive result about paths through computable manifolds. We write $\alpha \simeq \beta$ to denote that the paths α and β are homotopic.

Lemma 3.1. *Every loop f in a computable manifold M is homotopic to a computable loop there, and indeed is homotopic to a loop computed by an $\mathbb{R}$-computable function whose only real parameters are the base point of f and the parameters of the inclusion functions defining M.*

Proof. Simply to name an $\mathbb{R}$-computable loop homotopic to f is simple. The points t_i from Definition 1.6, and the values $g(t_i)$ and $h(t_i)$, altogether constitute finitely many real parameters, and the new loop just consists of a straight line in each chart $U_{g(t_i)}$ from the point $h(t_i)$ to the image in $U_{g(t_i)}$ of the point $h(t_{i+1}) \in U_{g(t_{i+1})}$. Since each $U_{g(t_i)}$ is homeomorphic to $\mathbb{R}^d$, it is simply connected, so clearly the old loop f and this new piecewise-linear loop are homotopic. The point of the lemma is to do something similar, but reducing the amount of nonuniform information required, so as to ensure that the new $\mathbb{R}$-computable loop can be specified by its base point and finitely much further information (namely rational numbers $\tilde{t}_i$ replacing the t_i, rational coordinates for the points $h(\tilde{t}_i)$, and the chart numbers $g(\tilde{t}_0), g(\tilde{t}_1), \ldots$). This will allow an $\mathbb{R}$-computable presentation of the fundamental group, given below, with finitely many real parameters: a single parameter which all by itself enumerates the rational coordinates and chart numbers of one computable path from each homotopy class, and finitely many more parameters to name the base point.

Let $f = \langle g, h \rangle$ be the original loop, with $0 = t_0 < t_1 < \cdots < t_n = 1$ as described in Definition 1.6. We first give the intuition for the proof, without which the details would be baffling. Each segment $h \restriction [t_m, t_{m+1}]$ of the loop f lies within a single chart $U_{g(t_m)}$, which is simply connected, being homeomorphic to $\mathbb{R}^d$. Within $U_{g(0)} \cap U_{g(t_1)}$ we pick an arbitrary point whose coordinates in $U_{g(0)}$ are all rational, and we define our computable loop $\overline{f}$ to begin with a line segment (in the $U_{g(0)}$ coordinates) from $h(0)$ to this point, given linearly in the variable $t \in [0, t_1]$. Continuing from this point, we add a new line segment (in the $U_{g(t_1)}$ coordinates) from there to a rational point in $U_{g(t_1)} \cap U_{g(t_2)}$, and so on, with the last line segment going from a point in $U_{g(t_{n-2})} \cap U_{g(t_{n-1})}$ back to the base point in $U_{g(1)} = U_{g(0)}$. By simple connectedness of each $U_{g(t_m)}$ and each $U_{k(g(t_m),g(t_{m+1}))} = U_{g(t_m)} \cap U_{g(t_{m+1})}$, we have $\overline{f} \simeq f$.

Notice that n and the points $t_1, \ldots, t_{n-1}$ are not given to us in any uniform way. However, since each $U_{g(t_m)} \cap U_{g(t_{m+1})}$ is open, its preimage under f is also open, and so we may assume that the points t_m have all been chosen (nonuniformly) to be rational. Now define $\overline{g} = g$, which is clearly $\mathbb{R}$-computable since g is piecewise constant. The key to the proof is our program for computing $\overline{h}$, the second component of the computable loop $\overline{f}$.

Using the density of $\mathbb{Q}^d$ in $\mathbb{R}^d$, we start by selecting a point $\vec{x}_1$ in $\mathbb{Q}^d \cap \text{range}(\varphi_{j(k(g(t_0),g(t_1)),g(t_0))})$, and defining $\overline{h} \restriction [t_0, t_1)$ to be the linear function from $\vec{x}_0 = h(0)$ to $\vec{x}_1$. Next, define $\overline{h}(t_1)$ to equal

$$\varphi_{j(k(g(t_0),g(t_1)),g(t_1))}(\varphi^{-1}_{j(k(g(t_0),g(t_1)),g(t_0))}(\vec{x}_1).$$

Since $\vec{x}_1 = \lim_{t \to t_1^-} \overline{h}(t)$, this allows $\overline{f}$ to satisfy the continuity requirement in the definition of path. Moreover, $\overline{h}(t_1)$ can be computed with no new real parameters: we use i, j, j', k, the base point $\vec{x}_0$, and the rational point $\vec{x}_1$.

We then continue inductively, picking an arbitrary point $\vec{x}_{m+1} \in \mathbb{Q}^d \cap \text{range}(\varphi_{j(k(g(t_m),g(t_{m+1})),g(t_m))})$, and defining $\overline{h} \restriction [t_m, t_{m+1})$ to be the linear function from $\overline{h}(t_m)$ to this point, then defining $\overline{h}(t_{m+1})$ to equal

$$\varphi_{j(k(g(t_m),g(t_{m+1})),g(t_{m+1}))}(\varphi^{-1}_{j(k(g(t_m),g(t_{m+1})),g(t_m))}(\vec{x}_{m+1}).$$

When we reach the case $m+1 = n$, of course, $t_{m+1} = 1$, so we no longer pick a rational point, but simply define $\vec{x}_n = \overline{h}(t_n) = \overline{h}(0)$, and let $\overline{h} \restriction [t_{n-1}, t_n]$ be the linear function from $\overline{h}(t_{n-1})$ to $\overline{h}(t_n)$. Thus we really do define a loop in M. Moreover, each line segment in the computable path $\overline{f}$ is homotopic to the corresponding segment $f \restriction [t_m, t_{m+1}]$, since each $U_{g(t_m)}$ is simply connected. This proves Lemma 3.1. $\square$

Theorem 3.2. *Let M be any $\mathbb{R}$-computable manifold which is connected but not simply connected. Then nullhomotopy of $\mathbb{R}$-computable loops in M is not $\mathbb{R}$-decidable: there is no $\mathbb{R}$-computable function ψ such that for every index $\vec{c}$ of a computable loop $f = \varphi_{\vec{c}}$ in M, $\psi(\vec{c})$ halts with output 1 if f is nullhomotopic in M, but halts with output 0 if f is not nullhomotopic in M.*

Of course, the converse is trivial: if M is simply connected, then nullhomotopy is decidable. Also, the hypothesis of connectedness of M is only to simplify the proof; for arbitrary M, any connected component M' can be presented as an $\mathbb{R}$-computable manifold, and if M' is not simply connected, then the theorem applies to M', hence gives the same result for M.

Proof. By assumption there is a loop in M, say with base point $\langle n, \vec{p}\rangle$, which is not nullhomotopic. By Lemma 3.1, it is homotopic to a computable loop $f = \langle g, h\rangle$ with the same base point. Using the Recursion Theorem for $\mathbb{R}$-computability (Theorem 1.2 above), for any $\mathbb{R}$-computable ψ, we define a computable function $\varphi_{\vec{e}}$ which "knows its own index $\vec{e}$" and can feed that index to ψ:

$$\varphi_{\vec{e}}(t) = \begin{cases} f((t-1)2^{s+1}+2), & \text{if } t \in [\frac{2^s-1}{2^s}, \frac{2^{s+1}-1}{2^{s+1}}) \ \& \ \psi(\vec{e})\downarrow = 1 \text{ in} \\ & \quad \text{exactly } s \text{ steps} \\ \langle n, \vec{p}\rangle, & \text{if not, or if } t = 1. \end{cases}$$

So this $\varphi_{\vec{e}}$ is a computable loop, and is the constant loop $\langle n, \vec{p}\rangle$ (hence nullhomotopic) unless $\psi(\vec{e})\downarrow = 1$. If convergence to 1 happens in exactly s steps, then $\varphi_{\vec{e}}$ is homotopic to f, since on the interval $[\frac{2^s-1}{2^s}, \frac{2^{s+1}-1}{2^{s+1}}]$ it copies the entire loop f, while staying constant everywhere else. Because f is not nullhomotopic, neither is this $\varphi_{\vec{e}}$. Thus ψ does not correctly decide nullhomotopy of $\varphi_{\vec{e}}$. □

Theorem 3.3. *Simple connectedness of $\mathbb{R}$-computable manifolds is not decidable by any $\mathbb{R}$-computable function. Specifically, there is no $\mathbb{R}$-computable ψ such that for every $\mathbb{R}$-computable M given by inclusion functions $i = \varphi_{\vec{c}}$, $j = \varphi_{\vec{d}}$, $j' = \varphi_{\vec{d'}}$, and $k = \varphi_{\vec{e}}$, $\psi(\vec{c}, \vec{d}, \vec{d'}, \vec{e})$ converges to 1 if M is simply connected, and converges to 0 if M is not simply connected.*

Proof. Again the Recursion Theorem is key. Fixing any $\mathbb{R}$-computable ψ, we define our parameters $\vec{c}$, etc., so that they start out by giving a basic non-simply-connected manifold, with $U_0 \cap U_1 = U_3$, $U_0 \cap U_2 = U_4$, $U_1 \cap U_2 = U_5$, and all other intersections empty. In particular, $U_0 \cap U_1 \cap U_2 = U_3 \cap U_2 = \emptyset$, so $\cup_{m \le 5} U_m$ is not simply connected. Then the program given by our parameters bides its time while running ψ on input $\langle \vec{c}, \vec{d}, \vec{d'}, \vec{e}\rangle$. If there is a stage s at which this computation halts and outputs 0, then our program adds a new chart U_s containing all of $\cup_{m \le 5} U_m$, thus making M simply connected. If there is no such s, then we never add any more new charts. So ψ does not decide simple connectedness of this M. □

We remark that a stronger statement is possible: simple connectedness is not decidable by any function of the form $\lim_s \theta(\vec{c}, \vec{d}, \vec{d'}, \vec{e}, s)$ with θ $\mathbb{R}$-computable. (By analogy to Turing computability, we might say that no $\mathbb{R}$-Δ^0_2 function decides simple connectedness.) One sees this by noting that in addition to the diagonalization strategy described above, we can extend a simply connected finite union of charts to a larger union which is not simply connected. So it is possible for M to flip back and forth every time θ changes its guess about the simple connectedness of M, forcing $\lim_s \theta$ to diverge.

We view these noncomputability results as reinforcing (albeit in a negative way) our general argument that $\mathbb{R}$-computable manifolds are best examined via Turing computation. Of course, Turing machines are fundamentally incapable of considering the question raised in Theorem 3.2 – nullhomotopy of $\mathbb{R}$-computable paths, and more generally, homotopy between two paths – because that question deals specifically with functions from the unit interval into $\mathbb{R}^\infty$. However, one might have hoped that $\mathbb{R}$-computation would suffice to decide such questions, in which case the model of $\mathbb{R}$-computation would have seemed more useful than the Turing model for dealing with these manifolds. Its failure to decide nullhomotopy, as shown in Theorem 3.2, voids this hope. $\mathbb{R}$-computation does not decide homotopy of paths, and if we consider paths only up to homotopy (as in the next section), then $\mathbb{R}$-computation will feel too strong, and Turing computation will suffice for examination of homotopy. This will also enable us to express simple-connectedness in a language accessible to Turing machines, by saying that a certain (Turing-)computably enumerable set contains all natural numbers.

4 Determining the Fundamental Group

In [5], the authors of this paper proved theorems showing that the fundamental group $\pi_1(M)$ of an $\mathbb{R}$-computable manifold M has its own $\mathbb{R}$-computable presentation, and that this presentation can be given uniformly up to the index for an $\mathbb{R}$-computable function s_M which depends on M. The following results appeared as Theorem 8, Proposition 1, and Corollary 1 there, with M assumed to be connected.

Theorem 4.1. *Let M be an $\mathbb{R}$-computable manifold. Then there is an $\mathbb{R}$-computable function S_M, defined on the naturals, such that the set $\{S_M(n) : n \in \omega\}$ consists of a set of indices for loops, and contains exactly one representative from each homotopy equivalence type.*

Proposition 4.2. *For every $\mathbb{R}$-computable manifold M, with S_M as in Theorem 4.1, there exists an $\mathbb{R}$-computable function $c_M : \omega \times \omega \to \omega$ which, on any input $\langle u, v \rangle$, outputs the unique w such that $S_M(u) * S_M(v) \simeq S_M(w)$.*

Corollary 4.3. *Let M be an $\mathbb{R}$-computable manifold. There is a uniform procedure to pass from an index for S_M to indices for an $\mathbb{R}$-computable presentation of the group $\pi_1(M)$ and for computing its word problem.*

The function S_M is a useful guide to our thinking about presenting fundamental groups, and the uniformity in the Corollary makes it clear that S_M contains all necessary information for an $\mathbb{R}$-computable presentation of $\pi_1(M)$. However, $\pi_1(M)$ is always countable (for second-countable manifolds), and $\mathbb{R}$-computation is simply not the appropriate model for dealing with functions from ω to ω, such as c_M above, or for presenting countable structures such as $\pi_1(M)$. Here we offer results along the same lines, on an arbitrary connected $\mathbb{R}$-computable manifold with inclusion functions i and k, but in the language of Turing computation, which we feel better reflects the computational issues involved. The actual function S_M from Theorem 4.1 cannot be formally given in this language, since it outputs indices for $\mathbb{R}$-computable loops $\langle g, h \rangle$, which may involve real parameters (e.g. those for the inclusion functions j and j' for M, as we must be able to identify points in the intersection $U_{k(g(t_m),g(t_{m+1}))}$ with points in $U_{g(t_m)}$ and in $U_{g(t_{m+1})}$, which requires using j and j'). However, we will give a representation of the elements of $\pi_1(M)$ by a set $G \subseteq \omega^{<\omega}$ modulo an enumerable equivalence relation $\sim$, all of which will use Turing computation below i and k. The index for S_M above plays essentially the same role above as the $\sim$-oracle does for us, and the c_M corresponding to our representation will in fact be Turing-computable, outputting a representative of the concatenation of the two loops, modulo $\sim$. The uniform procedure described in Corollary 4.3 will now be uniform in a $\sim$-oracle. Thus, Corollary 4.3 can essentially be proven from our constructions below, noting that every function from $\omega^{<\omega}$ to ω is in fact $\mathbb{R}$-computable, and that the relation $\sim$, which is enumerable computably in $(i \oplus k)$ and can be encoded as a single real parameter, allows computation of the word problem for $\pi_1(M)$.

The following topological lemma will be necessary. The connectedness conditions in the lemma are satisfied by any $\mathbb{R}$-computable manifold; we state them to make clear exactly what assumptions are used in the proof.

Lemma 4.4. *Let α and β be paths from point a to point b in a topological manifold M covered by three open sets U_m, U_n, and U_p. Assume that $a \in U_m \cap U_p$ and $b \in U_n \cap U_p$, and that β lies entirely within U_p, while α has an initial segment contained in U_m and the remainder contained in U_n. Moreover, assume that U_m, U_n, and U_p are all simply connected, and that their pairwise intersections are path-connected. Then $\alpha \simeq \beta$ iff $U_m \cap U_n \cap U_p \neq \emptyset$.*

Proof. The forward direction is essentially topology, of course, and we omit the details here. For the converse, suppose there exists a point $x \in U_m \cap U_n \cap U_p$, and let γ be a path from a to x lying within $U_m \cap U_p$, and δ a path from x to b within $U_n \cap U_p$. Also, fix t such that $\alpha(t) \in U_m \cap U_n$, and let θ be a path from x to $\alpha(t)$ lying within $U_m \cap U_n$. (All this is possible by the assumption of connectedness of intersections.) Then

$$\alpha \simeq (\alpha\restriction[0,t]) * \theta^{-1} * \theta * (\alpha\restriction[t,1]) \simeq \gamma * \delta$$

since γ and $(\alpha\restriction[0,t] * \theta^{-1})$ both lie within U_m, hence are homotopic, and likewise $\delta \simeq (\theta * \alpha\restriction[t,1])$ within U_n. On the other hand, $(\gamma * \delta)$ is contained within U_p, hence is homotopic to β. (So we have used simple-connectedness of all three charts.) This proves $\alpha \simeq \beta$. □

The next definition largely explains how we will present the fundamental group of M using Turing programs with oracles for the functions i and k.

Definition 4.5. Let M be a topological manifold covered by charts U_m (for m in some index set I). We say that a path α through M is *given by* a finite string $\sigma \in I^{<\omega}$ if there exist finitely many real numbers $0 = t_0 < t_1 < \cdots < t_{1+|\sigma|} = 1$ such that for all $m < n$, the image of $\alpha\restriction[t_m, t_{m+1}]$ is contained in $U_{\sigma(m)}$.

Of course, a single path may be given by many distinct strings σ. Lemma 4.7 will shed light on this case.

Theorem 4.6. *There exist oracle Turing programs Φ, Ψ, and Γ such that, for any connected $\mathbb{R}$-computable manifold M given by inclusion functions i, j, j', and k, the program Φ^i computes the characteristic function of a set G, the program $\Psi^{i\oplus k}$ enumerates an equivalence relation $\sim$ on G, and Γ (with no oracle) computes a total binary operation F on G, compatible with $\sim$, under which $(G/\!\sim)$ is isomorphic to the fundamental group $\pi_1(M)$.*

Proof. Since M is connected, we may select any base point in M and consider $\pi_1(M)$ to be the group of loops in M with that base point, modulo homotopy, under concatenation. Fix this base point b_0 to be the image in U_0 of the origin in $\mathbb{R}^d$, under α_0. G will represent the space of all loops with that base point (with a certain amount of homotopy already taken into account – so that G can be countable, for one thing), $\sim$ will be the rest of the homotopy relation on G, and F will be the operation of concatenation. We now give details.

The set G is a subset of $\omega^{<\omega}$. To lie in G, a string σ must satisfy:

$$\sigma(0) = \sigma(|\sigma| - 1) = 0 \ \& \ (\forall p < |\sigma|) i(\sigma(p), \sigma(p+1)) \neq -1.$$

We view this σ as representing a loop which begins in $U_{\sigma(0)} = U_0$ and continues into $U_{\sigma(1)}$, then $U_{\sigma(2)}$, etc. The conditions above simply say that such a loop does exist in M: each chart in the sequence intersects the next one. Also, it says that the base point is b_0, at least modulo a trivial loop within U_0. Clearly this G is uniformly computable in an i-oracle. Moreover, the concatenation function F on G really is just concatenation, hence computable:

$$F(\sigma, \tau) = \langle \sigma(0), \ldots, \sigma(|\sigma| - 1), \tau(0), \ldots, \tau(|\tau| - 1) \rangle.$$

The basic operation for the relation $\sim$ enumerated by $\Psi^{i \oplus k}$ comes from Lemma 4.4. We start by declaring that if $i(k(m, n), p) \neq -1$, then every $\sigma \in G$ of the form

$$\langle \sigma(0), \ldots, \sigma(t-1), m, n, \sigma(t+2), \ldots, \sigma(|\sigma| - 1) \rangle$$

is related by $\sim$ to

$$\langle \sigma(0), \ldots, \sigma(t-1), m, p, n, \sigma(t+2), \ldots, \sigma(|\sigma| - 1) \rangle$$

provided that the latter string also lies in G. (So each of $i(m, n)$, $i(m, p)$, and $i(p, n)$ is $\neq -1$.) Notice that the case $m = p$ shows that $\sim$ always allows an entry in σ to be replaced by two consecutive instances of itself. We then close $\sim$ under reflexivity, symmetry, and transitivity, so that it is an equivalence relation computably enumerable below an oracle for the functions i and k.

Intuitively $\sim$ says that if the path passes through U_m and then through U_n, and $U_m \cap U_n$ intersects U_p, then we may deform the path so that it passes from U_m into U_p and thence into U_n. (Of course, passing from U_m into U_p need not imply leaving U_m, so it is possible that $U_p \subseteq U_m$, or that similar relations hold.) Conversely, we can eliminate p from the string $\langle \ldots, m, p, n, \ldots \rangle$ giving this path if Lemma 4.4 allows it, i.e. if $(U_m \cap U_n)$ intersects U_p.

Lemma 4.7. *Suppose that a path α through an $\mathbb{R}$-computable manifold is given by each of the two strings σ and τ in $\omega^{<\omega}$. Then $\sigma \sim \tau$.*

Proof. Let $\mathcal{U}$ be the finite set of charts containing all U_m with $m \in \text{range}(\sigma) \cup \text{range}(\tau)$ and closed under intersections (i.e. under the function k). Then there is a string $\rho \in \omega^{<\omega}$ such that α is given by ρ and range(ρ) is contained in the set

$$\{k(m,n) : m \in \text{range}(\sigma)\ \&\ n \in \text{range}(\tau)\}.$$

Now we claim that if $\rho = \langle \rho(0), \ldots, p, k(m,n), q, \ldots, \rho(|\rho|-1)\rangle$, we have both

$$\rho \sim \langle \rho(0), \ldots, p, m, q, \ldots, \rho(|\rho|-1)\rangle \text{ and}$$
$$\rho \sim \langle \rho(0), \ldots, p, n, q, \ldots, \rho(|\rho|-1)\rangle.$$

(The lemma then follows easily by induction.) To see the first of these, here are the steps, all of which follow the basic rule for $\sim$:

1. $\langle \ldots, p, k(m,n), q, \ldots\rangle$
2. $\sim \langle \ldots, p, k(m,n), k(m,n), q, \ldots\rangle$
3. $\sim \langle \ldots, p, k(m,n), m, k(m,n), q, \ldots\rangle$
4. $\sim \langle \ldots, p, m, k(m,n), q, \ldots\rangle$ (since $i(k(p,m), k(m,n)) \neq -1$)
5. $\sim \langle \ldots, p, m, q, \ldots\rangle$ (since $i(k(m,q), k(m,n)) \neq -1$).

To see why line 4 (and similarly line 5) are allowed, notice first that since $\rho \in G$, we know that $i(p, k(m,n)) \neq -1$. Then we use the different conditions of Definition 1.3. Now $i(k(p,k(m,n)), k(m,n)) = 1$, by Condition 2, and we claim that $i(k(p,k(m,n)), k(m,p)) = 1$ also. (Condition 3 will then force $i(k(p,m), k(m,n)) \neq -1$, as needed.) The claim follows from Lemma 1.4 and symmetry of the function k: we know $k(p,k(m,n)) = k(k(m,p),n)$, so $i(k(p,k(m,n)), k(m,p)) = i(k(k(m,p),n), k(m,p)) = 1$ by Condition 2. $\square$

Now we arrive at the heart of the proof: the claim that this relation $\sim$ is equal to the relation of homotopy, when we understand $\sigma \in G$ to represent a loop beginning in $\sigma(0)$ and passing through each $\sigma(t)$ in turn. Lemma 4.4 ensures that $\sim$ is a subset of the homotopy relation. It is also clear that every loop α in M with base point b_0 has a (non-unique) representative in G, arising from any finite sequence of charts which together (in order) cover

α. So we must show that if σ and τ in G represent homotopic loops α and β, then $\sigma \sim \tau$ under our definition above.

Let $F : [0,1]^2 \to M$ be a homotopy from α to β, so $F(x,0) = \alpha(x)$, $F(x,1) = \beta(x)$, and $F(0,y) = F(1,y) = b_0$ for all x and y. Since F is continuous, its image in M is compact, hence covered by a finite collection of simply connected charts (without loss of generality say $U_0, \ldots, U_N$), which we may take to be closed under intersections. We also assume that N is chosen so that σ and τ assume only values $\leq N$.

Now every loop of the form $F_{y_0}(x) = F(x, y_0)$, for any fixed y_0, can be expressed by some finite string $\sigma_{y_0} \in (N+1)^{<\omega}$ of charts through which it passes. Moreover, there will be an open interval $I_{y_0} \subseteq [0,1]$ containing y_0 such that every path F_y with y in that interval can be expressed by the same finite string σ_{y_0}. By compactness, then, there will be finitely many values, say $0 = y_0 < y_1 < \cdots < y_p = 1$, such that the entire unit interval is covered by the open intervals I_{y_t} around these values. For $y \in I_{y_t} \cap I_{y_{t+1}}$, F_y can be expressed by both strings σ_{y_t} and $\sigma_{y_{t+1}}$, and since $\{U_0, \ldots, U_N\}$ is closed under intersection, Lemma 4.7, or finitely many uses of the two defining properties of $\sim$, can be used to show that $\sigma_{y_t} \sim \sigma_{y_{t+1}}$. By transitivity, then, we will have $\sigma = \sigma_{y_0} \sim \sigma_{y_p} = \tau$. This completes the proof that $\sim$ is in fact the homotopy relation on G, and it is clear now that the fundamental group $\pi_1(M)$ is given by $(G/\!\sim)$ under concatenation F. □

It follows from these results that the question of simple connectedness of a manifold can also be expressed over ω. Specifically, M is simply connected iff the equivalence relation $\sim$ constructed above turns out to contain all pairs in the i-computable subset $(G \times G)$ of ω^2. Since $\sim$ is computably enumerable below an oracle for $(i \oplus k)$, simple connectedness is a Π^0_2 property of the reals i and k.

5 Conclusions and Questions

Section 2 and Theorem 4.6 are the backbone of our contention that the questions addressed in this paper are truly questions to be asked using Turing computation, rather than $\mathbb{R}$-computation. The $\mathbb{R}$-computable aspect of our Definition 1.3 is the collection of functions $\varphi_{j(m,n)}$ and their inverses $\varphi_{j'(m,n)}$, and these simply do not enter into the discussion of paths and homotopy. Of course, we do not claim that fundamental groups must be Turing-computable:

they require oracles for the functions i and k of the manifold, and even then, the presentation of the fundamental group only gives it as a quotient of a computable set by a c.e. equivalence relation, which was seen in Theorem 3.2 not to be uniformly $\mathbb{R}$–decidable in general, let alone decidable by a Turing machine (with or without an $(i \oplus k)$-oracle). Theorem 4.1, Proposition 4.2, and Corollary 4.3 represent a way of presenting $\pi_1(M)$ $\mathbb{R}$-computably and isolating the (countably much) information necessary to do so, but we consider Theorem 4.6 to be the essential solution to the question.

Of course, Definition 1.3 was quite strict, requiring that an $\mathbb{R}$-computable manifold be presented as a set of simply-connected charts closed under intersection. Classically this does not normally present obstacles, but from the point of view of effectiveness, there could be considerable difficulty in passing from a presentation of charts not satisfying this requirement to a presentation which does satisfy it. Indeed, there would likely be no small difficulty just in determining whether a presentation satisfies the requirement or not. It is quite possible that here $\mathbb{R}$-computability might come to the fore, displacing Turing computability.

For those desiring true interaction between $\mathbb{R}$-computability and manifolds, however, we strongly recommend consideration of analytic properties, rather than topological ones. Any notion of placing a metric on a manifold, or of finding geodesics, seems very likely to require an $\mathbb{R}$-computable function on each chart, defining the distance metric there – or alternatively, an $\mathbb{R}$-computable presentation of the tangent bundle of the manifold, allowing one to determine the length of a path by integration. We leave these investigations for another paper, with encouragement for all interested researchers to consider them.

In Section 2, we noted that every second-countable manifold covered by a countable atlas of simply-connected charts, closed under intersection, must have inclusion functions of the form described in Definition 1.3, except that the functions $\varphi_{j(m,n)}$ may not be $\mathbb{R}$-computable. (One should not call them $\varphi_{j(m,n)}$ in this case, of course; the $\varphi_{\vec{e}}$ notation is intended to denote $\mathbb{R}$-computable functions.) Following the terminology of computable model theory, we might refer to any i, k, and $\{\psi_{m,n} : i(m,n) = 1\}$ as a *presentation* of a manifold M if they satisfy the conditions of Definition 1.3 (with $\psi_{m,n}$ and its inverse in place of $\varphi_{j(m,n)}$ and $\varphi_{j'(m,n)}$) and give rise to a manifold homeomorphic to M. The *Turing degree* of the presentation would then be the Turing degree of $(i \oplus k)$, and the $\mathbb{R}$-*degree* of the presentation should be the $\mathbb{R}$-degree of the partial map $\langle m, n\rangle \mapsto \langle \psi_{m,n}, \psi_{m,n}^{-1}\rangle$, to the extent that this can be expressed in $\mathbb{R}$-degree theory. (Of course, i and k, being functions on the naturals, would contribute nothing to this $\mathbb{R}$-degree.) One would then naturally consider $\mathbb{R}$-degrees of distinct presentations of the same manifold. A natural first question is whether there exists a second-countable manifold which is homeomorphic to no $\mathbb{R}$-computable manifold. This appears to be a challenging question.

(The results in this paper should allow one to produce a manifold M which is homeomorphic to no manifold M' with Turing degree $\mathbf{0}$ and with arbitrary $\mathbb{R}$-degree. Just make the fundamental group of M sufficiently complicated as to have no presentation as a computable set with computable binary operation modulo a c.e. equivalence relation. Any M' homeomorphic to M would have this same fundamental group, and Theorem 4.6 would then show that the functions i' and k' used to present M' could not both be computable.)

Of course, all of these same questions can be addressed using other models of computation on $\mathbb{R}$. This might require writing a paper such as this one all over again, using the other model; or it might have substantial parallels to the present paper. We would be interested to know which is the case, as such investigations could help spotlight similarities and differences between the two models of computation. Computable analysis in particular seems likely to bear fruit in this way, given appropriate definitions for the basic concepts. We have focused this paper specifically on $\mathbb{R}$-computability, but we very much hope that this focus is not taken as an intention to exclude other models of computation from consideration. Our basic conjecture is that results using computable analysis would follow the same lines as ours here: questions about homotopy and the fundamental group would likely reduce to Turing computability, but more analytic questions, about connections and metrics on manifolds and the like, are potentially very fruitful.

References

[1] L. Blum, F. Cucker, M. Shub & S. Smale, *Complexity and Real Computation* (Springer, 1997).

[2] L. Blum, M. Shub, & S. Smale, On a theory of computation and complexity over the real numbers, *Bulletin of the American Mathematical Society (New Series)* **21** (1989), 1–46.

[3] E.H. Brown, Computability of Postnikov Complexes, *Annals of Mathematics* **65** (1957), 1–20.

[4] W. Calvert, On three notions of effective computation over $\mathbb{R}$, to appear in *Logic Journal of the IGPL*.

[5] W. Calvert & R. Miller, Real Computable Manifolds and Homotopy Groups, in *Unconventional Computation - Eighth International Conference, UC 2009*, eds. C.S. Calude, J.F. Costa, N. Dershowitz, E. Freire, & G. Rozenberg, *Lecture Notes in Computer Science* **5715** (Springer-Verlag: Berlin, 2009), 98–109.

[6] W. Calvert & J.E. Porter, $\mathbb{R}$-computable structure theory, preprint.

[7] Ю. Л. Ершов, Определимость и Вичислимость, Сибирская Школа Алгебры и Логики (Научная Книга, 1996).

[8] J.D. Hamkins, R. Miller, D. Seabold, & S. Warner, An introduction to infinite time computable model theory, in *New Computational Paradigms*, eds. S.B. Cooper, B. Löwe, & A. Sorbi (Springer, 2008) 521–557.

[9] A. Hatcher, *Algebraic Topology* (Cambridge University Press: Cambridge, 2001).

[10] G. Hjorth, B. Khoussainov, A. Montalban and A. Nies, From automatic structures to Borel structures, preprint (2008).

[11] R. Miller, Locally computable structures, in *Computation and Logic in the Real World - Third Conference on Computability in Europe, CiE 2007*, eds. B. Cooper, B. Löwe, & A. Sorbi, *Lecture Notes in Computer Science* **4497** (Springer-Verlag: Berlin, 2007), 575–584.

[12] А. С. Морозов, Элементарные подмодели параметризуемых моделей, Сибирский Математический Журнал **47** (2006), 595–612.

[13] R.I. Soare; *Recursively Enumerable Sets and Degrees* (Springer-Verlag: New York, 1987).

[14] A. Weil, Review of *Introduction to the theory of algebraic functions of one variable*, by C. Chevalley, *Bulletin of the American Mathematical Society* **57** (1951), 384–398.

[15] H. Weyl, *The concept of a Riemann surface*, 3rd ed. (Addison-Wesley, 1955)

Department of Mathematics and Statistics
Murray State University
Murray, KY 42071 U.S.A.
E-mail: Wesley.Calvert@murraystate.edu
URL: http://campus.murraystate.edu/academic/faculty/wesley.calvert

Department of Mathematics
Queens College – C.U.N.Y.
65-30 Kissena Blvd.
Flushing, New York 11367 U.S.A.
Ph.D. Programs in Mathematics & Computer Science
C.U.N.Y. Graduate Center
365 Fifth Avenue
New York, New York 10016 U.S.A.
E-mail: Russell.Miller@qc.cuny.edu
URL: http://qcpages.qc.cuny.edu/~rmiller