A Constructive Formalization of the Fundamental Theorem of Calculus

Luís Cruz-Filipe

Department of Computer Science, University of Nijmegen, The Netherlands
Center for Logic and Computation, IST, UTL, Portugal
lcf@cs.kun.nl

Abstract. We have finished a constructive formalization in the theorem prover Coq of the Fundamental Theorem of Calculus, which states that differentiation and integration are inverse processes. In this formalization, we have closely followed Bishop's work ([4]). In this paper, we describe the formalization in some detail, focusing on how some of Bishop's original proofs had to be refined, adapted or redone from scratch.

1 Introduction

In this paper we describe how constructive real analysis can be formalized in the theorem prover Coq. The results proved include notions of continuity, differentiability, integration and the main theorems in calculus of one variable: Taylor's Theorem and the Fundamental Theorem of Calculus.

This formalization was developed using the algebraic hierarchy developed for the FTA project described in [14] and extending it whenever necessary. Working in this way, we intend to show that it is possible to formalize large pieces of mathematics in a modular way—that is, such that new blocks can be built on top of the old ones without disrupting the existing work. We feel that this is the way to successfully build a comprehensive library that can be actually used in real mathematics.

We assume all functions to be partial so that we can define all usual functions of real analysis (e.g. the logarithm). For this, we identify domains with their characteristic predicate and represent each real-valued function by a binary function which takes a proof term as a second argument—a proof that the function is defined at the point where we are trying to evaluate it. This process, which is described in detail in [7], is very similar to the approach which was originally followed in the Automath system (see for example [3]). Of course, total functions simply correspond to the case when the predicate is always true.

For generality's sake, we decided to work constructively (following Bishop, see [4]). This means essentially that we do not in general accept reasoning by contradiction and work with an equality which is not decidable. On the other hand, we make no extra assumptions about the properties of real numbers, which means that within classical mathematics (that is, where the axiom $(A \vee \neg A)$ is accepted) our work is still valid. Arguments on why our approach may be argued to be more sensible can be found in the first chapter of [4].

In this paper, we will follow the structure of Chapter 2 of [4] (which was the reference closely followed throughout the whole formalization) and compare several of the statements and proofs therein with those in our formalization, focusing specifically on the two or three points where we had to follow a different path than his and trying to understand why this was so. We feel that it is relevant to point out that those were the exceptional cases—most of the formalization amounted in fact to choosing good representations for the definitions, translating Bishop's original proofs into `Coq` code and filling in the details—, meaning that the formalized work is an accurate representation of the piece of informal mathematics we started with.

A relevant part of the work which we will not go into in this paper is automation. The interested reader can find more information about some automation techniques that were developed in parallel with the formalization in [7].

The formalization itself, including some documentation, can be downloaded from `http://www.cs.kun.nl/~lcf/ftc.tgz`.

2 Basic `Coq` Notation

This paper intends to focus on the mathematical aspects of the formalization, rather than in specific `Coq` issues. However, we will present some specific `Coq` terms; for the reader unfamiliar with the `Coq` syntax, we briefly present the notations we will need:

- There are two basic types for data types and propositions, which are respectively `Set` and `Prop`[1];
- λ-abstraction is denoted by square brackets; therefore, `[x:A]M` represents the term $\lambda x : A.M$;
- Π-abstraction (and universal quantification) is denoted by curved brackets; therefore, the term `(x:A)M` corresponds to the term $\Pi x : A.M$ or, via the Curry-Howard isomorphism, to the proposition $\forall x : A.M$;
- Existential quantification and Σ-types are represented by curly braces; for example, the term `{x:A & M}` can correspond to the term $\Sigma x : A.M$ or to the proposition $\exists x : A.M$;
- The logical connectives $\wedge$ (and) and $\vee$ (or) are represented respectively by `*` and `+`[2].
- The usual algebraic operations (addition, multiplication, etc.) and relations (less, less or equal, equal) will be denoted by their usual symbols enclosed in square brackets: `[+]`, `[*]`, `[<=]` and so on. In particular, `[--]` denotes the unary group inverse operator.

More specific notation will be explained whenever it occurs.

[1] A more precise description of these types and of the `Coq` type theory can be found in the `Coq` reference manual, see [6], but is not needed to understand this presentation.

[2] This is not standard `Coq` notation; a more detailed explanation for these connectives can be found in [14].

3 The Real Numbers, Sequences and Series

The algebraic hierarchy which constituted the basis for our work already included a definition of real numbers. This definition does not completely coincide with Bishop's, so we will briefly discuss both constructions and compare them.

In the FTA project, a real number structure `IR` was defined axiomatically as being a complete ordered field with the archimedian property, that is, an ordered field with an operation `lim` such that (1) every Cauchy sequence s converges to $\texttt{lim}(s)$ and (2) for every natural number n, there is an element x of the field such that $x < \underline{n}$ (where $\underline{n}$ denotes the image of n in the ring).[3]

Then, a concrete structure, the set of Cauchy sequences of rational numbers with equality defined as equality of limits, is defined (see [13]) and proved both to satisfy these axioms and to be isomorphic to every other structure that satisfies these axioms. However, in our work we suppose an arbitrary real number structure. This allows us to be more general and apply it to any other such structure, and is also more efficient as the construction of the concrete structure does not need to be loaded into memory.

Bishop takes a slightly different approach, defining a real number to be a regular sequence[4] of rational numbers. Two such sequences $\{x_n\}$ and $\{y_n\}$ correspond to the same real number iff their difference converges to 0 in the rational numbers.

Sections 1 and 2 of [4] amount mainly to establishing that this structure is indeed a real number structure in the sense above defined; from this it follows that it is isomorphic to every other real number structure, and therefore all of the remaining work applies to any such structure.

We decided to work in the more general setting of real number structures for two reasons. On the one hand, we wanted to use the work which previously had been done for the FTA project; on the other hand, we felt that working with an axiomatic characterization would lead to more generality, as we then can apply our results not only to Bishop's real numbers but to any other real number structure without any further work.

Sequences and Series

Section 2.3 of [4] is concerned with properties of sequences and series of real numbers.

In the FTA project, sequences were already quite extensively treated, covering most of the reference material. These include a predicate `Cauchy_prop` that states that a sequence is a Cauchy sequence (see [14]); in a real number structure there is also an operator `Lim` that associates to every Cauchy sequence its limit. New results include a straightforward definition of subsequence and its main properties, which we will not discuss.

[3] This axiomatization is based on Heyting's work on algebraic structures presented in [19].

[4] A regular sequence is a Cauchy sequence such that $\forall_{m,n\in\mathbb{N}}\ |x_m - x_n| \leq \left|\frac{1}{m} - \frac{1}{n}\right|$

To study series, we begin by associating to each sequence the sequence of its partial sums in the obvious way:

```
Definition seq_part_sum [x:nat->IR] := [n:nat](sum0 n x).
```

where `(sum0 n x)` simply represents $\sum_{i=0}^{n-1} x_n$.

Following Bishop, a series is said to converge iff this sequence is a Cauchy sequence; in this case, the limit of this sequence is said to be the sum of the series. The formalization of these is direct, using the definitions already present for sequences:

```
Definition convergent [x:nat->IR] :=
  (Cauchy_prop (seq_part_sum x)).
Definition series_sum [x:nat->IR][H:(convergent x)] :=
  (Lim (seq_part_sum x) H).
```

Two criteria are then proved for determining convergence of a series: the comparison test and ratio test (respectively Propositions 9 and 10 in Chapter 2 of [4]).

As an example, we discuss the formalization of the comparison test. In Bishop's book, this reads:

Proposition: If $\sum_{n=1}^{\infty} y_n$ is a convergent series of nonnegative terms and $|x_n| \leq y_n$ for each n, then $\sum_{n=1}^{\infty} x_n$ converges.

In this statement, the hypothesis that y is nonnegative is superfluous; therefore, we formalize this lemma simply as

```
Lemma comparison : (x,y:nat->IR)(convergent y)->
  ((n:nat)(AbsIR (x n))[<=](y n))->(convergent x).
```

To use this lemma in practice, it is useful to weaken the hypothesis in the last result further and use the following

```
Lemma str_comparison : (y:nat->IR)(convergent y)->
  {k:nat & ((n:nat)(le k n)->(AbsIR (x n))[<=](y n))}->
  (convergent x).
```

which only requires that $|x_n|$ be bounded by y_n from some point on.

The ratio test is similarly stated and proved both in Bishop's formulation and in a similar generalized way.

As special cases, e and π are defined as the some of two series. For the formalization, this is done in three steps: first, we define the relevant sequence; then, we prove it converges as a series; finally, we define the constant in terms of the sum of this series. As an example, e is defined by $e = \sum_{n=0}^{\infty} \frac{1}{n!}$, which is formalized in the following three steps:[5]

[5] In the first definition, the notation is simplified, as we are omitting a proof term—namely, one which states that $n! \neq 0$.

```
Definition e_series := [n:nat]One[/](fac n).
Lemma e_series_conv : (convergent e_series).
Definition E := (series_sum e_series e_series_conv).
```

4 Continuous Functions

When looking at properties that real-valued functions may have, it is usual to do so in two levels: point-wise (f has property P at x) or in an interval (f has property P on I, meaning that f has property P at x for all $x \in I$). If P is a property characterized by an ε-δ definition (that is, it is of the form $\forall_\varepsilon \exists_\delta$), then there is also a corresponding *uniform* notion obtained by a quantifier interchange. Classically, if I is compact it is usually a theorem that having property P on I is equivalent to having property P uniformly on I.

For example, letting P be the property of "being continuous", the point-wise definition (at an arbitrary point x) reads

$$\forall_{\varepsilon>0}\exists_{\delta>0}\forall_{y\in\mathbb{R}}|x-y|<\delta\rightarrow|f(x)-f(y)|<\varepsilon\ .$$

The corresponding global definition on an interval I would then read

$$\forall_{\varepsilon>0}\forall_{x\in I}\exists_{\delta>0}\forall_{y\in I}|x-y|<\delta\rightarrow|f(x)-f(y)|<\varepsilon\ .$$

And finally the uniform definition is

$$\forall_{\varepsilon>0}\exists_{\delta>0}\forall_{x,y\in I}|x-y|<\delta\rightarrow|f(x)-f(y)|<\varepsilon\ ,$$

and these last two are proved to be equivalent for closed finite I.

Constructively, however, things happen in a different way. This is mainly for two reasons: first, equality on the real numbers is undecidable, and thus point-wise information about f is seldom usable; on the other hand, uniform properties are not implied by point-wise ones on closed finite intervals (because these cannot be constructively proved to be compact in the classical sense).

Bishop gets around this problems by defining a compact interval to be a finite closed interval[6]. He then argues that uniform concepts are the only relevant ones, and defines the corresponding global ones (in an arbitrary interval I) as follows: f has property P in I if for every compact interval $[a, b] \subseteq I$ it is the case that f has property P in $[a, b]$.

For formalization purposes, this requires two levels of reasoning to be always present. At a lower level, properties are defined in a compact interval—which consists simply of two real numbers a and b and a proof that $a \leq b$. Then the corresponding global property is defined in terms of the local one. We give the example of continuity: first, in a context where `a,b:IR`, `a[<=]b` holds, `f` is a partial function, `I` is the predicate characterizing $[a, b]$ and P is the predicate characterizing the domain of f[7], we define

[6] From now on, when we speak of a *compact* interval, we will be referring to a closed, finite interval. When we want to speak about a set satisfying the classical definition of compactness we will speak of a *classically compact set.*

[7] Subsets are in fact identified with predicates, see [7]

```
Definition continuous_I := (included I P)*
  ((e:IR)(Zero[<]e)->{d:IR & (Zero[<]d)*
    ((x,y:IR)(I x)->(I y)->((AbsIR x[-]y)[<=]d)->
      ((AbsIR (f x)[-](f y))[<=]e))}).
```

A few comments are due: all functions are assumed to be partial, so there is an extra assumption at the beginning that $[a, b]$ is within the domain of f; this also means that in fact we can't simply write down `(f x)`—there needs to be some proof term, but for clarity of exposition (and as its form is irrelevant from the mathematical point of view) we chose to omit it from this presentation. Finally, this definition differs from the classical one in that the inequalities are stated with $\leq$ instead of $<$; this is because, within the constructive framework, statements about $\leq$ can be proved by contradiction (because $x \leq y$ is defined as $\neg(y < x)$). It is easily seen, however, that these definitions are equivalent.

Because we work constructively, the value of `d` can be effectively computed from `e`; we will call this operation the *modulus of continuity* for f and, when needed, we will denote it by ω.

In order to make the general definition, we need to be able to speak about intervals. The approach we took was the following: we defined a syntactic type of intervals and a function `iprop` that associates to each of them a predicate. For example, the real line is associated to $\lambda x : \mathbb{R}$. `True`; the interval $]a, b]$ is associated to $\lambda x : \mathbb{R} \,.\, a < x \wedge x \leq b$, and so forth. This allows us to write down the following definition, where `PartIR` is the type of partial functions:

```
Definition Continuous [I:interval][f:PartIR] :=
  (included (iprop I) (Pred f))*
  ((a,b:IR)(Hab:a[<=]b)(included (compact a b Hab) (iprop I))->
    (continuous_I a b f)).
```

We now prove, closely following [4], that both notions of continuity are preserved through algebraic operations: addition, subtraction, multiplication, division, composition and absolute value of continuous functions all yield continuous functions. However, unlike in the classical case, some side conditions have to be assumed:

- In the case of multiplication, we have to assume that the values of the functions considered are bounded. This turns out to be a general property of continuous functions in a compact interval, stated as the Corollary of Theorem 3 in Chapter 2 of [4]; its proof, however, is not as trivial as therein indicated, and we will discuss it hereafter.
- For composition, given continuous $f : I \to \mathbb{R}$ and $g : J \to \mathbb{R}$, we can prove $g \circ f$ to be continuous provided, obviously, that $f(I) \subseteq J$, but furthermore we need to assume that the image of every compact subinterval of I is contained in a compact subinterval of J. Although this is classically the case, constructively it is not provable.[8]

[8] This may seem a bit counterintuitive; the problem is that, constructively, given a compact $A \subseteq I$, we cannot compute a maximum (respectively minimum) of $f(A)$,

– A consequence of the previous remark is that the rule for division also has a side condition—namely, that the denominator function g be not only non-zero but bounded away from zero, that is, for some $c > 0$ it is the case that $|g(x)| \geq c$ for all x in the relevant interval. Once again, classically this is trivially true.

In order to bound the value of a continuous function in a compact interval, Bishop starts by making the following definition:

Definition: A set $A \subseteq \mathbb{R}$ is said to be *totally bounded* iff for every $\varepsilon > 0$ there exist points $x_1, \ldots, x_n$ such that for every $y \in A$ one of the numbers $|y - x_1|, \ldots, |y - x_n|$ is less than ε.

The reason for this definition is the following: classically, being compact is equivalent to being totally bounded and complete; however, constructively this is not true, and in particular closed intervals cannot be constructively proved to be classically compact. However, they *are* totally bounded and complete, and that is enough to prove the usual results in analysis.

Bishop now proves that every totally bounded set has a least upper bound and a greatest lower bound. Finally, it is shown that the image of a compact set through a continuous function is totally bounded.

In formalizing this reasoning there turn out to be two major problems. The first one is a technical issue: the definition of totally bounded can be written down, given a set A, as

$$\forall_{\varepsilon>0}\exists_{n\in\mathbb{N}}\exists_{x_1,\ldots,x_n}\forall_{y\in A}\exists_{i\in\{1,\ldots,n\}}|y - x_i| < \varepsilon \ .$$

But formalizing this definition cannot be done in a direct way, as we have a variable number of existential quantifiers. We manage to get around this problem by quantifying over a list:

```
Definition totally_bounded [P:IR->Set] : Set :=
  (e:IR)(Zero[<]e) -> {l:(list IR) &
    ((x:IR)(member x l)->(P x))*
      (x:IR)(P x)->{y:IR & (member y l)*((Abs x[-]y)[<]e)}}.
```

The problem, however, is that the introduction of lists is quite unnatural and generates some complexity which is usually not present in the original (informal) proofs.

The second problem is that Bishop assumes without proving that a compact interval is totally bounded. This is actually the case, and it is probably quite obvious to anyone; formalizing it, however, requires giving an algorithm to determine, given the points $x_1, \ldots, x_n$ and y, an index i such that x_i is close enough

but only a least upper bound (resp. greatest lower bound), which is not guaranteed to be actually in the image of f, and may therefore lie outside of J. For a model of Bishop style mathematics where a function which *doesn't* satisfy this property see Theorem 8.1 on p. 71 of [2]

to y. This is achieved through dividing the interval $[a, b]$ into n subintervals of length $\frac{\varepsilon}{2}$ (appealing to the archimedian axiom); then, using a tricky induction argument and the properties of the less than relation, we find an i such that $a_i < y < a_{i+2}$, from which it is easy to prove that $|y - a_{i+1}| < \varepsilon$.

Finally, we define (uniform) convergence of sequences and series of functions in compact and general intervals and prove convergence criteria for series analogous to the ones for real number series; we define the limit (or sum) of a sequence (or series) of continuous functions and prove continuity of the thus defined function. As a special (and very important) case, we define power series and prove the Dirichlet criterion for the interval of convergence of such series. These formalizations are very close both to those in [4] and to the similar proofs for sequences and series of real numbers, and are therefore quite straightforward.

5 Differentiation

The formalization of differential calculus follows Bishop closely. As before, we define first what it means for a function f' to be the derivative of f in a proper[9] compact interval $[a, b]$ with characteristic predicate I:

```
Definition derivative_I [f,f':PartIR] :=
  (continuous_I a b f)*(continuous_I a b f')*
  ((e:IR)(Zero[<]e)->{d:IR & (Zero[<]d)*
    (x,y:IR)(I x)->(I y)->((AbsIR x[-]y)[<=]d)->
      (AbsIR (f y)[-](f x)[-](f' x)[*](y[-]x))[<=]e[*](AbsIR y[-]x)}).
```

Then, we define the general concept in an arbitrary proper interval:

```
Definition Derivative [I:interval][pI:(proper I)][f,f':PartIR] :=
  (included (iprop I) (Pred f))*(included (iprop I) (Pred f'))*
  ((a,b:IR)(Hab:a[<]b)(included (compact a b Hab)) (iprop I))->
    (derivative_I a b f f')).
```

In both cases, the requirement that the interval is proper (that is, it contains more than one point) is important to assure uniqueness of the derivative.

At first sight, the constructive definition may seem a bit different from the classical one; but in fact it isn't. Classically they are equivalent; however, writing down the division requires the existence of some proof terms, which means that this definition is constructively more general: the classical version gives us no information regarding what happens for values of x and y in I such that it is not known whether or not $x = y$.

We then follow Bishop closely and prove the usual rules for derivation of sums, products, powers, quotients and composition of differentiable functions. In the last two cases, we assume side conditions similar to the ones assumed to prove preservation of continuity.

[9] E.g. not empty.

The most important results in this section are the constructive versions of Rolle's theorem, the Mean Law and Taylor's theorem. These three theorems differ from their classical counterparts in a similar way: classically, they state the existence of a point, under suitable hypotheses, satisfying a certain equality; constructively, they state that given any positive number ε there exists a point which satisfies the same equality up to an error smaller than ε.

For example, the constructive version of Rolle's theorem reads as following:

Theorem: Let f be differentiable on the interval $[a, b]$ and let $f(a) = f(b)$. Then for each $\varepsilon > 0$ there exists x in $[a, b]$ with $|f'(x)| \leq \varepsilon$.

This is formalized in the following way:

```
Theorem Rolle : (a,b:IR)(Hab:a[<]b)(f,f':PartIR)
  (derivative_I a b f f')->((f a)[=](f b))->
  (e:IR)(Zero[<]e) -> {x:IR & (I x)*(AbsIR (f' x))[<=]e}.
```

The proof of this result follows Bishop's with no significant modifications. As a straightforward corollary, we get the constructive Mean Law:

Theorem: Let f be differentiable on the interval $[a, b]$. Then for each $\varepsilon > 0$ there exists x in $[a, b]$ with $|f(b) - f(a) - f'(x)(b - a)| \leq \varepsilon$.

The formalization of Taylor's theorem requires a number of auxiliary notions to be defined prior to it. However, it gives no new insights into the process of formalization, being quite similar (though more complicated) to these two examples; therefore, we will not discuss it here.

Some general remarks are due on the statements of these theorems. Being presented as approximations, they are at first sight not as useful as their classical counterparts; however, in most applications the presence of an equality doesn't really help, as it holds for an unspecified existentially quantified point in a compact interval, and the best we can get is an inequality. Taking the Mean Law as an example, the only thing it allows us to establish without question is that

$$|f(b) - f(a)| \leq ||f'||_{[a,b]} * |b - a| \ ,$$

where $||f'||_{[a,b]}$, the norm of f' in $[a, b]$, is the least upper bound of the image of $[a, b]$ through $|f'|$, and this will be the practical application of the theorem.

Interestingly, this formulation is valid both classically and constructively (classically it is immediate; constructively it can also be proved simply by observing that if we add any positive constant to the righthandside then we have an upper bound for the expression on the lefthandside). The fact that some authors state and prove it directly in this form (Dieudonné is one of them, see for example [8] and [9]) is evidence that at least for some people it is the best formulation of the Mean Law.

6 Integration

Integration turned out to be by far the most difficult process to formalize following Bishop's work. There were several reasons for this:

- The need for heavy computation involving sums—in previous work we had already come across several computations and majorations, but they required usually little more than properties of the absolute value and algebraic identities;
- The need for very technical lemmas which include specific identities between sums, results about proof irrelevance and formalizing fuzzy concepts like "sufficiently close approximation".

In this section, we will outline the process of definition of the integral and focus on the proof of one specific lemma, which accidentally was incorrect in the reference book, and which illustrates quite well the kind of technicalities that are needed at the level of formulation—as well as the kind of proof steps that don't seem likely to be automated in the near future.

Following Bishop, we define a variant of the Riemann integral of continuous functions. There are two main reasons for this choice, namely:

- The classical construction of the Riemann lower and upper integrals as lower and upper bounds of sequences of sums cannot directly be made constructive, as those bounds are not guaranteed to exist; adding the assumption that the function we are integrating is continuous, however, allows us to prove constructively that they do exist and coincide, yielding a simple definition;
- On the other hand, all constructive functions are believed to be continuous, which means that this approach does not imply less generality; in other words, we can integrate every function we can define, so the need to look at other integrals is not so imperative.

Finally, we also chose to follow the Bishop formalization in order to be able to compare our work with his. Therefore, we did not consider alternative definitions which might be easier to formalize; we hope to look at those in a near future.

We begin by defining a *partition* of a compact interval $[a, b]$ with length n as a finite sequence $(a_0, \ldots, a_n)$ such that $a_0 = a$, $a_n = b$ and for $0 \leq i < n$ it is the case that $a_i \leq a_{i+1}$. We formalize this as a record type:

```
Record partition [a,b:IR][Hab:a[<=]b][lng:nat] : Set :=
  {pts    : (i:nat)(le i lng)->IR;
   prf1   : (i,j:nat)i=j->(Hi:(le i lng))(Hj:(le j lng))
              (pts i Hi)[=](pts j Hj);
   prf2   : (i:nat)(H:(le i lng))(H':(le (S i) lng))
              (pts i H)[<=](pts (S i) H');
   start  : (H:(le O lng))(pts O H)[=]a;
   finish : (H:(le lng lng))(pts lng H)[=]b}.
```

This is simply the `Coq` way to say that a partition (that is, an element of type `partition`) is a 5-tuple; each component of the tuple has an identifier which allows us to refer to it and a type. Thus, `pts` is the function that given i provides the point a_i; `prf2`, `start` and `finish` contain proof terms that ensure that the required properties hold. The proof term (second argument) in `pts` is required because we only want to have $n+1$ points in the partition[10]; `prf1` states that this proof term does not influence the choice of the points.

The *mesh* of a partition is the greatest of the values $a_{i+1} - a_i$. A partition $Q = (a'_0, \ldots, a'_m)$ is said to be a *refinement* of P iff for every i there is a j such that $a_i = a'_j$; in other words, P is a subsequence of Q.

Given a partition P of $[a, b]$, if $x_0, \ldots, x_{n-1}$ satisfy the condition $a_i \leq x_i \leq a_{i+1}$ we say that they *respect* the partition P. This is formalized by a predicate `points_in_partition`. Given a function f, a number S_P of the form

$$S_P = \sum_{i=0}^{n-1} f(x_i)(a_{i+1} - a_i)$$

is said to be a sum of f that respects P. We can then define this sum in the following way, where for clarity we are omitting the proof terms and some of the types:

```
Definition partition_sum [P,x,f][H:(points_in_partition P x)] :=
  (Sum [i:nat](f (x i))[*]((pts P (S i))[-](pts P i))).
```

Given any partition $a_0, \ldots, a_n$ there is a canonical choice for x: just take $x_i = a_i$; also, given any interval $[a, b]$ and a natural number n there is a canonical partition of $[a, b]$: just take $a_i = a + i\frac{b-a}{n}$ (called an *even partition* of $[a, b]$). This justifies that we define the canonical sequence of sums, given a function f, as the sequence of canonical sums of the even partitions with increasing number of points:

$$S(f, n) = \sum_{i=0}^{n-1} f\left(a + i\frac{b-a}{n}\right)\frac{b-a}{n} \ .$$

```
Definition integral_seq : nat->IR :=
  (even_partition_sum a b f (S n)).
```

In order to prove convergence of this sequence, the following theorem is needed:

Theorem: Let f be a continuous function on a compact interval $[a, b]$ with modulus of continuity ω. If P is any partition of $[a, b]$, if $\varepsilon > 0$, and if $\text{mesh}(P) \leq \omega(\varepsilon)$, then, for any sum S_P of f respecting P, there is an n such that

$$|S_P - S(f, n)| \leq \varepsilon(b - a) \ . \tag{1}$$

[10] We could have required instead that $a_m = b$ for $m > n$, but this actually makes things harder, so we chose to keep closer to the original definition.

The proof of this result relies on the following two lemmas, where it is assumed that f is a continuous function with modulus of continuity ω in $[a, b]$:

Lemma 1: If P and Q are partitions of $[a, b]$, $\text{mesh}(P) \leq \omega(\varepsilon)$ and Q is a refinement of P, then, for any sums S_P and S_Q of f respecting, correspondingly, P and Q, we have that $|S_P - S_Q| \leq \varepsilon(b - a)$.

Lemma 2: If P and R are partitions such that $\text{mesh}(P) \leq \omega(\varepsilon)$ and $\text{mesh}(R) \leq \omega(\varepsilon')$, and if there exists a partition Q which is simultaneously a refinement of P and of R, then for any sums S_P and S_R of f respecting P and R it is the case that $|S_P - S_R| \leq (\varepsilon + \varepsilon')(b - a)$.

The proof of the first result presents no problems other than technical issues. It hangs mainly on the following fact: given i, we know that there are j and j' such that $a_i = a'_j$ and $a_{i+1} = a'_{j'}$; this allows us to write S_P in terms of points of Q, and use the modulus of continuity to establish the result. However, some manipulation of double sums is required which involves a lot more than just trivial computation.

The second lemma is quite simpler, as it just amounts to two applications of the first one: $|S_P - S_R| = |(S_P - S_Q) + (S_Q - S_R)|$, and from the triangle inequality we get the expected result.

The theorem can then be proved assuming every two partitions have a common refinement and applying the second lemma to S_P and the sequence $S(f, n)$; using properties of inequalities and limits we arrive at the required result. All that remains is proving that any two partitions share a common refinement, which is stated without proof in [4]

Unfortunately, though classically this is a trivial statement, constructively it is not true! The reason for that is that in a partition points must be ordered, and $\leq$ is not decidable on the real numbers.[11]

This error was corrected in [5] in the following way: first, we say that two partitions $P = (a_0, \ldots, a_n)$ and $R = (b_0, \ldots, b_m)$ are separated iff for all i and j in the appropriate ranges $a_i < a_{i+1}$ and $b_j < b_{j+1}$; furthermore, if $0 < i < n$ and $0 < j < m$ then $a_i \neq b_j$.

Now, we can prove that any two separated partitions have a common refinement. This is a trivial consequence of co-transitivity of the $<$ relation: we can always tell, for every i and j, that either $a_i < b_j$ or $b_j < a_{i+1}$, which allows us to order the points.[12] The theorem is then proved by taking close enough approximations of P and R that are separated.

Of course, though we can intuitively see that the "close enough approximations" exist, to prove the result we have to construct them; we will now explain how this is done[13].

[11] This is easy to see, as equality can be expressed in terms of $\leq$ by the relation $x = y$ iff $x \leq y \wedge y \leq x$.

[12] Formalizing this, though not complex, is still a long and tedious process.

[13] This is a level of detail to which [4] never goes, and in our view really illustrates the difference between formal and informal mathematics.

We formalize the property of separation in two steps. A partition is $P = (a_0, \ldots, a_n)$ said to be (simply) separated iff $a_i < a_{i+1}$:

```
Definition separated [P:(Partition a b n)] :=
  (i:nat)(Pts P i)[<](Pts P (S i)).
```

Two partitions are said to be (mutually) separated iff each of them is separated and if $a_i \neq b_j$ whenever $0 < i < n$ and $0 < j < m$.

```
Definition Separated [P:(Partition a b n)]
  [Q:(Partition a b m)]:= (separated P)*(separated Q)*
  (i,j:nat)(lt O i)->(lt O j)->(lt i n)->(lt j m)->
    (Pts P i)[#](Pts Q j).
```

As before, we have omitted some proof terms in these definitions.

The construction of separated approximations of two partitions is done in two steps. First, given a partition P and positive real numbers α and ξ, we want an algorithm to get a separated partition P' with the following properties:

- $\text{mesh}(P') \leq \text{mesh}(P) + \xi$;
- for every sum S_P respecting P we can find a sum $S_{P'}$ respecting P' such that $|S_P - S_{P'}| < \alpha$.

To do this, we take δ to be $\min(\xi, \frac{\alpha}{n \cdot M})$, where n is the number of points in P and M is the norm of f in $[a, b]$. δ is positive, which means that for every real number x either $x > 0$ or $x < \frac{\delta}{2}$. We then recursively define the following sequence of partitions:

- $P^0 = P$;
- P^{i+1} is obtained from P in the following way: for every pair a^i_j, a^i_{j+1} of consecutive points in P^i, test whether $a^i_{j+1} - a^i_j > 0$ or $a^i_{j+1} - a^i_j < \frac{\delta}{2}$. If there is a j for which the second is the case, choose the least such j and define $a^{i+1}_m = a^i_m$ for $m \leq j$ and $a^{i+1}_m = a^i_{m-1}$ for $m > j$[14] (that is, obtain P^{i+1} by removing the $(j+1)$th point in P^i); else $P^{i+1} = P^i$.

This construction always gets to a fixed point, provided $b - a$ is sufficiently big (which is OK, as if b and a are too close the theorem holds trivially). This is a partition P' satisfying both desired conditions (the first is trivial; for the second, take any choice of points respecting P and simply remove the points corresponding to points that were removed in P').

Now, given P and R, we determine separated partitions P' and R' by the above construction; then, we shift the points in P' by a similar (but even less obvious) construction to get a partition P'' which is also separated from R' and for which the previous two properties hold.

At this point, there turns out to be still a small detail which has to be corrected in the statement of the Theorem. We assumed that we began with a

[14] This is a slight simplification, as we have to take some care if $j + 1$ is the length of P^i, but we won't go into that level of detail here.

partition P with $\text{mesh}(P) \leq \omega(\varepsilon)$; however, although we can take the approximations with mesh as close to P as we want, we cannot actually require them to be equal (to see this, consider the case when P is an even partition; then any shifting of its points will necessarily increase the mesh). This invalidates the reasoning through approximations, as if $\text{mesh}(P') > \omega(\varepsilon)$ we can no longer establish a bound for the sum. We solve this problem by requiring (in the statement of the theorem) that $\text{mesh}(P) < \omega(\varepsilon)$. We can then find approximations which still respect that inequality (just take $\alpha = \frac{1}{2}\omega(\varepsilon) - \text{mesh}(P)$), and we are still able to apply lemma 2.

It is then trivial to prove that the sequence of sums we previously defined is a Cauchy sequence; the integral of f in $[a, b]$ is defined as its limit.

```
Lemma Cauchy_integral_Seq : (Cauchy_prop integral_seq).

Definition integral := (Lim integral_seq).
```

Linearity and monotonicity of the integral operator are proved simply by unfolding the definition of integral and appealing to the corresponding properties of limits of Cauchy sequences and of sums.

It has been pointed out that all the problems we discussed arose simply because we have a definition of partition which is too general; in fact, to define the integral we only need even partitions, so we could simply have restricted our attention to these. This would simplify matters a lot, as it is trivial to define a common refinement of any two even partitions, and we wouldn't need all these auxiliary concepts.

Up to this point, this is indeed true; and our first approach upon stumbling with the above-mentioned error in Bishop's original proof was to restrict our attention to even partitions. Unfortunately, for the next result (which is a fundamental theorem, and not just an auxiliary lemma) we really need the general definition, and at this stage we had to go back and redo our work according to [5].

We want to show that

$$\int_a^b f(x)dx = \int_a^c f(x)dx + \int_c^b f(x)dx \tag{2}$$

whenever $a \leq c \leq b$. This is trivially done using properties of limits, closely following Bishop's proof, and appealing to (1). This requires choosing arbitrary (even) partitions of $[a, b]$ and $[b, c]$ and obtaining from those a partition of $[a, c]$ which contains all the points in the two original partitions. However, if we take for example $a = 0$, $b = 1$ and $c = \sqrt{2}$ it is easy to see that there can be no even partition of $[a, c]$ which refines even partitions on $[a, b]$ and $[b, c]$, except in trivial cases. Therefore, we really must consider partitions in general.

With these considerations in mind, we prove (2) and use that as a motivation to define, for arbitrary a and b, $\int_a^b f(x)dx = \int_{\min(a,b)}^b f(x)dx - \int_{\min(a,b)}^a f(x)dx$:

```
Definition Integral :=
  (integral (Min a b) b f)[-](integral (Min a b) a f).
```

As usual, we slightly simplified the `Coq` code by omitting some proof terms.

It is easy to prove that this new integral inherits all the properties of the old one. We finally define an operator `FPrim` that takes as arguments a function f, an interval I, a point $a \in I$ and a proof that f is continuous in I and yields the primitive of f defined by $g(x) = \int_a^x f(t)dt$. This is a continuous function, and we can prove the fundamental theorem of calculus:

Theorem: Let f be a continuous function on a proper interval I and $a \in I$. Let g be the function defined in I by the expression $g(x) = \int_a^x f(t)dt$; then:

1. f is a derivative of g in I;
2. if f is a derivative of g_0 in I, then the difference $g - g_0$ is a constant function in I;
3. for every g_0 such that f is a derivative of g_0 in I and for every points $x, y \in I$, $\int_x^y f(t)dt = g_0(y) - g_0(x)$.

These theorems are formalized as follows: first, we take any proper interval `I` and function `f` continuous in `I`; we let `a` be a point of `I` and define `g:=(FPrim f a)`.

We first state that f is a derivative of g in I.

```
Theorem FTC1 : (Derivative I g f).
```

We now take any other `g0` and assume that `f` is a derivative of `g0` in `I`, that is, that there exists a term of type `(Derivative I g0 f)`. We can now prove:

```
Theorem FTC2 : {c:IR & (Feq I g{-}g0 {-C-}c)}.
```

Here, `Feq` is a ternary relation that states that the second and third arguments (functions) coincide in the domain given as first argument, `{-}` is a notation for function subtraction, and `{-C-}c` denotes the constant function with value c. This theorem thus states that there is a real number c such that in I g and g_0 differ by c.

Finally, the last part of the theorem is stated as follows:

```
Theorem FTC3 : (x,y:IR)(iprop I x)->(iprop I y)->
  (continuous_I (Min x y) (Max x y) f)->
  (Integral x y f)[=](g0 b)[-](g0 a).
```

Interestingly, the formalization of the proofs of the first two results requires little more than what is presented in [4]. The third part of the theorem is not presented there, but it is the usual classical formulation of the FTC.

As corollaries of this theorem, we are able to prove that if $\{f_n\}$ is a sequence of continuous functions converging uniformly to a continuous function f then both the sequence of derivatives $\{f_n'\}$ and the sequence of primitives $\{F_n\}$ with the same origin will converge respectively to the derivative f' and the primitive F of f, assuming all these exist.

7 Transcendental Functions

To conclude this work, Bishop defines some of the most important functions in analysis and proves their main properties using the tools previously built. We will briefly show how this work was formalized.

The exponential, sinus and cosine are all defined as power series. Using the previously established results, we defined an operator `FPowerSeries` that assigns to every real number sequence a a sequence of functions defined by

$$f_n(x) = \frac{a_n}{n!}(x - x_0)^n \ ,$$

where x_0 is a parameter. We then prove that under suitable conditions the sum of these functions is defined for all real numbers.

To define the exponential function, we take $x_0 = 0$ and $a_n = 1$; we can then easily prove that this series converges in the real line.

```
Definition Exp_ps := (FPowerSeries Zero [n:nat]One).

Lemma Exp_conv : (fun_series_convergent_IR realline Exp_ps).
```

Next, the partial function `Expon` is defined as the sum of this series; this function is total, so we define `Exp:IR->IR` as an abbreviation so that we can always forget about proof terms:

```
Definition Expon:=(FSeries_Sum Exp_ps Exp_conv).
Definition Exp := [x:IR](Expon x Set_I).
```

Here `Set_I` is a canonical inhabitant of `True`, which is the predicate for the domain of `Expon`.

The definition of sinus and cosine is very similar; the tangent is then defined as the quotient of these two functions.

As for the logarithm, it is defined in the interval $]0, \infty[$ as the indefinite integral with origin 1 of the function defined by $f(x) = 1/x$. We begin by proving that this function is continuous in that interval, and define the logarithm as the corresponding indefinite integral (recall the definition of `FPrim` in the previous section):

```
Lemma log_defn_lemma : (Continuous (openl Zero) {1/}FId).
Definition Log := (FPrim (openl Zero) {1/}FId log_defn_lemma One).
```

The inverse trigonometric functions are similarly defined.

We finish with a small selection of the main results we have proved. We would like to point out that no proof terms have been omitted in what follows—it is completely correct `Coq` code.

- Algebraic properties: the equation $e^{x+y} = e^x e^y$ is formalized in `Coq` as the term `(x,y:IR)(Exp x[+]y)[=](Exp x)[*](Exp y)`;

- Order properties: $e^x > 0$ reads `(x:IR)(Zero[<](Exp x))`;
- Inverse relation properties: the fundamental relation $e^{\log(x)} = x$ is expressed as `(x:IR)(H:Zero[<]x)(Exp (Log x H))[=]x`.
- Analytical properties: the following result states that `Expon` is the only function that is its own derivative and evaluates to 1 at 0:

```
Lemma Exp_unique : (F:PartIR)(Derivative realline Set_I F F)->
  (H1:(Pred F Zero))(F Zero H1)[=]One->
    (Feq (iprop realline) Expon F).
```

The proofs of the majority of these results are extremely simple, and amount basically to translating the proofs in [4] to `Coq` commands and proving eventual trivial side conditions.

Trigonometric functions and their inverses are defined in a similar way; their basic properties are then proved just as those for the exponential and logarithmic function were, and for conciseness we will not present them.

8 Related Work

Several formalizations of real numbers, real analysis and properties of elementary transcendental functions have been previously completed in different systems. They all differ from ours in that they are classical formalizations, however, and they have not taken constructive issues into account.

Mizar [1] presently includes a classical formalization of real analysis. Differental calculus was developed by J. Kotowicz, K. Raczkowski and P. Sadowski, whereas N. Endou, K. Wasaki, and Y. Shidama have formalized integral calculus. The classical counterparts to the results which we presented are all included in this formalization; it is also interesting to note that it is the only other formalization of those here mentioned that explicitly attempts to deal with partial functions.

Micaela Mayero has formalized differential calculus and transcendental functions in Coq, starting with an axiomatic characterization of the reals, and showed how this formalization can be used to prove correctness of programs in numerical analysis (see [20] and [21]).

John Harrison [17] has also formalized real numbers and differential calculus on his HOL-light system. This has been used together with his formalization of floating point arithmetic, described in [18], to prove correctness of floating point algorithms

Similarly, Bruno Dutertre has developed a library of real analysis (see [10]) which was later extended by Hanne Gottliebsen to include the elementary transcendental functions and their properties. Gottliebsen proceeds to show in [16] how this system can be used interactively with computer algebra systems to ensure (greater) correctness of the results obtained by these.

On the other hand, work has been done on exact real number arithmetic. Some representations of real numbers are presented and briefly discussed by A. Edalat and P. J. Potts in [11]; Edalat and Krznaric further show in [12]

how one specific representation can be used to compute integrals. It would be interesting to examine how well these real number representations fit with our axiomatization of the reals, but we feel that that would be outside of the scope of this paper.

9 Conclusions

As we have showed, we successfully managed to formalize a significant piece of mathematics, namely the chapter on real analysis of [4] which corresponds quite closely to a basic course on real analysis at undergraduate level. In doing so, we feel to have provided evidence for the claim that it is possible to formalize large pieces of mathematics that can actually be used.

The modular way in which the formalization was done also showed that it is possible to build large libraries which can be built and consistently enlarged: as we mentioned, we worked using the algebraic library which was developed for and extensively used in the FTA project [14]; in the end, we obtained a much larger library without having to change any of its original content.

We did not discuss automation in this paper, as it was already done in [15] and [7]. In those papers, it was shown how several frequently occurring goals – including proofs of algebraic identities and checking that a function is continuous – can be automatically solved or, at least, significantly simplified. However, the work on integration underlined the need for a much higher level of automation, which may probably be efficiently achieved only through communication with computer algebra systems, as described in [22]. Still, we feel that this work is a significant step toward the building of a useful library of formalized analysis that can be actually used in the building of interactive proofs.

Finally, we feel to have given further arguments favoring Bishop's claim that the constructive way to do things is at least as powerful as the classical one, as we proved the most important results of real analysis. Our proofs, being constructive, have the advantage of possessing computational content, which in theory allows the `Coq` extraction mechanism to generate from them algorithms to actually compute with real numbers. This hasn't been actually done, as the memory and time requirements demanded are currently too high for it to be feasible; however, we feel we have shown the way in which it can actually be done.

Acknowledgments

Support for this work was provided by the Portuguese Fundação para a Ciência e Tecnologia, under grant SFRH / BD / 4926 / 2001 and by the FCT and FEDER via CLC.

The author would also like to thank H. Barendregt, H. Geuvers, B. Spitters and F. Wiedijk both for the many discussions throughout the development of this work, which contributed to its successful outcome, and for their suggestions regarding the contents and form of this paper.

References

1. `http://www.mizar.org`
2. Beeson, M., *Foundations of constructive mathematics*, Springer-Verlag, 1985
3. Benthem Jutting, L. S. van, *Checking Landau's "Grundlagen" in the Automath System*, in Nederpelt, R. P., Geuvers, J. H. and de Vrijer, R. C. (Eds.), *Selected Papers on Automath*, North-Holland, 1994
4. Bishop, E., *Foundations of Constructive Analysis*, McGraw-Hill Book Company, 1967
5. Bishop, E. and Bridges, D., *Constructive Analysis*, Springer-Verlag, 1985
6. The Coq Development Team, *The `Coq` Proof Assistant Reference Manual Version 7.2*, INRIA-Rocquencourt, December 2001
7. Cruz-Filipe, L., *Formalizing Real Calculus in `Coq`*, in *Theorem Proving in Higher Order Logics*, Carreño, V., Muñoz, C. and Tahar, S. (eds.), NASA Conference Proceedings, Hampton, VA, 2002
8. Dieudonné, J., *Foundations of Modern Analysis*, Academic Press, New York, 1969
9. Dieudonné, J., *Calcul Infinitésimal*, Hermann, Paris, 1968
10. Dutertre, B., *Elements of Mathematical Analysis in PVS*, 9th International Conference, TPHOLs 1996, Springer LNCS 1125, 1996
11. Edalat, A. and Potts, P. J., *A New representation for Exact real Numbers*, in *Electronic Notes in Theoretical Computer Science* vol. 6, 1997
12. Edalat, A. and Krznaric, M., *Numerical integration with Exact Arithmetic*, in *Proceedings of ICALP'99*, 1999
13. Geuvers, H. and Niqui, M., *Constructive Reals in `Coq`: Axioms and Categoricity*, in Callaghan, P., Luo, Z., McKinna, J. and Pollack, R. (Eds.), *Proceedings of TYPES 2000 Workshop*, Durham, UK, LNCS 2277
14. Geuvers, H., Pollack, R., Wiedijk, F. and Zwanenburg, J., *The Algebraic Hierarchy of the FTA Project*, in Linton, S. and Sebasitani (eds.), *Journal of Symbolic Computation, Special Issue on the Integration of Automated Reasoning and Computer Algebra Systems*, pp. 271-286, Elsevier, 2002
15. Geuvers, H., Wiedijk, F. and Zwanenburg, J., *Equational Reasoning via Partial Reflection*, in *Theorem Proving in Higher Order Logics*, 13th International Conference, TPHOLs 2000, Springer LNCS 1869, 162-178, 2000
16. Gottliebsen, H., *Transcendental Functions and Continuity Checking in PVS*, in *Theorem Proving in Higher Order Logics*, 13th International Conference, TPHOLs 2000, Springer LNCS 1869, 197-214, 2000
17. Harrison, J., *Theorem Proving with the Real Numbers*, Springer-Verlag, 1998
18. Harrison, J., *A machine-checked theory of floating point arithmetic*, in *Theorem Proving in Higher Order Logics*, 12th International Conference, TPHOLs 1999, Springer LNCS 1690, 113-130, 1999
19. Heyting, A., *Intuitionism: an Introduction*, Studies in Logic and the Foundations of Mathematics, North-Holland Publishing Company, Amsterdam, 1956
20. Mayero, M., *Formalisation et automatisation de preuves en analyses réelle et numérique*, PhD thesis, Université Paris VI, décembre 2001
21. Mayero, M. *Using Theorem Proving for Numerical Analysis*, in *Theorem Proving in Higher Order Logics*, 15th International Conference, TPHOLs 2002, Springer LNCS 2410, 246-262, 2002
22. Oostdijk, M., *Generation and Presentation of Formal Mathematical Documents*, Ph.D. Thesis, Technische Universiteit Eindhoven, 2001