Exploring Euler's Foundations of Differential Calculus in Isabelle/HOL using Nonstandard Analysis

Jessika Rockel

Master of Science
Computer Science
School of Informatics
University of Edinburgh
2019

Abstract

When Euler wrote his ‘Foundations of Differential Calculus’ [5], he did so without a concept of limits or a fixed notion of what constitutes a proof. Yet many of his results still hold up today, and he is often revered for his skillful handling of these matters despite the lack of a rigorous formal framework.

Nowadays we not only have a stricter notion of proofs but we also have computer tools that can assist in formal proof development: Interactive theorem provers help users construct formal proofs interactively by verifying individual proof steps and providing automation tools to help find the right rules to prove a given step.

In this project we examine the section of Euler's ‘Foundations of Differential Calculus’ dealing with the differentiation of logarithms [5, pp. 100-104]. We retrace his arguments in the interactive theorem prover Isabelle to verify his lines of argument and his conclusions and try to gain some insight into how he came up with them.

We are mostly able to follow his general line of reasoning, and we identify a number of hidden assumptions and skipped steps in his proofs. In one case where we cannot reproduce his proof directly we can still validate his conclusions, providing a proof that only uses methods that were available to Euler at the time.

Acknowledgements

I would like to thank my supervisor, Dr. Jacques Fleuriot, for all of his support and guidance throughout this process. Working with him was a delight and an incredible learning experience for me. His expertise, feedback, encouragement and humour were much appreciated.

Many thanks also to Marlene Compton and Lukas Domagala for their unending support and hard work proofreading this dissertation.

Table of Contents

Chapter 1

Introduction

In his 'Foundations of Differential Calculus' [5], Euler establishes a lot of facts about derivatives that hold up to this day. But while these days it is most common to use limits to reason about derivatives, the concept of limits did not exist at the time. Euler argues his case using infinitely small and infinitely large numbers and defines differentials: infinitely small changes in a function $y = f(x)$ resulting from infinitely small changes dx in the input x, which he defines as $dy = f(x+dx) - f(x)$. He later uses these to calculate the ratio of the differential to the change in the input, arriving at a form $\frac{dy}{dx}$ that is similar to the widely used modern definition of the derivative based on limits.

Throughout the book, he uses this structure to explore differentials and derivatives of many different types of functions. We have chosen to examine the section on the differentiation of logarithms in chapter 6, *On the Differentiation of Transcendental Functions*, using the translation by John D. Blanton [5, pp 100-104]. It is the first section of that chapter, immediately following the chapter *On the Differentiation of Algebraic Functions of One Variable* [5, pp. 77-98], which deals with more trivial functions. As such it is one of the first sections to present results with challenging proofs while not building too much on results from previous chapters.

The aim of this project is to capture Euler's reasoning formally in the interactive theorem prover Isabelle, with the goal of both verifying (or refuting) his conclusions and also identifying gaps in his reasoning and the hidden assumptions those point to. Similar things have been done for other historical texts [15], other works of Euler's [12], and another section of this book [9], but the section we have chosen has not been formalised before.

We use Robinson's nonstandard analysis [18] as a framework for interpretation of Euler's use of infinitely small and infinitely large numbers. Like limits, it is a concept developed to give a consistent and rigorous backing to the way Euler, Leibniz and Newton reasoned about the infinitely small and large. Nonstandard analysis has been formalised in Isabelle [8], which the previous formalisations of Euler's work [12, 9] also built upon.

In chapters 2 and 3 we will provide some historical background and go through the technical details necessary to understand the project. Chapter 4 presents the proofs we formalised, including details about the methods used and problems encountered, followed by our conclusions in chapter 6.

Chapter 2

Context

In this section, we want to provide some context regarding Euler's methods in relation to mathematicians of the same time period and how his work is viewed today. We will also touch on the practice of using interactive theorem provers to formalise historical mathematical texts.

2.1 Historical Background

While in his writing, Euler mostly only references his own statements from previous books, the main ideas he was working with in his 'Foundations of Differential Calculus'[5] were not entirely new: Leibniz and Newton are widely accepted to be the ones to (independently) discover the differential calculus and the first to systematically use infinitely small quantities to do calculus [10]. Euler used a similar idea, first in his 'Introduction to the Analysis of the Infinite' [4], where he also presented infinite series definitions for logarithmic, exponential and trigonometric functions, and later in his 'Foundations of Differential Calculus', which includes an entire chapter detailing his assumptions about infinitely small and infinitely large numbers. As we will touch on in the next chapter, his assumptions are generally consistent, however, they are in many places quite vague or only correct with some leeway for interpretation.

Mathematicians such as Cauchy and Weierstrass have since then worked on developing a more rigorous system that captures the ideas Newton, Leibniz and Euler had about infinitesimal changes in values, their effects on functions and their use in differentiation. This most famously resulted in the conception of limits, which are used to define derivatives to this day. They offer a way to avoid explicitly introducing infinitely small and large numbers by instead saying that the value of a given variable tends towards zero or infinity respectively.

In the introduction to his book ‘Non-Standard Analysis’ [18] presenting his theory of the same name, Abraham Robinson positions nonstandard analysis as an alternative to limits that more faithfully captures Leibniz’s ideas about infinitesimals. He explicitly extends the real numbers to include infinitesimals and infinitely large numbers and provides a proof of the *transfer principle*, stating that any appropriately formulated statement is true of $^*\mathbb{R}$ (the extended field of *hyperreal* numbers) if and only if it is true of $\mathbb{R}$. We will provide a short introduction to nonstandard analysis in chapter 3.

2.2 Previous Analyses of Euler’s Proofs

Mathematicians and math historians alike often praise Euler’s intuitive understanding of the concepts he is describing while at the same time criticising the lack of rigour in his proofs. This is evidenced for example by this paragraph taken from ‘An Introduction to the History of Mathematics’ by Howard and Jamie H. Eves:

> It is perhaps only fair to point out that some of Euler’s works represent outstanding examples of eighteenth-century formalism, or the manipulation, without proper attention to matters of convergence and mathematical existence, or formulas involving infinite processes. He was incautious in his use of infinite series, often applying to them laws valid only for finite sums. Regarding power series as polynomials of infinite degree, he heedlessly extended to them well- known properties of finite polynomials. Frequently, by such careless approaches, he luckily obtained truly profound results... [6, p.435]

On the other hand, Euler is frequently revered for his handling of infinitesimals. Gordon, Kusraev and Kutateladze [11] position Euler as far ahead of his time, describing how it took mathematicians a while to switch from pedantically pointing out Euler’s perceived mistakes to catching up with his ideas when it comes to infinitesimal analysis. In the review ‘Is Mathematical History Written by the Victors?’, the authors express a similar view, stating that “Euler used infinitesimals *par excellence*, rather than merely ratios thereof, in a routine fashion in some of his best work” [1, p. 894].

Detlef Laugwitz was the first to write about what he calls “hidden lemmas” present in Euler’s proofs (and those of Fourier, Poisson and Cauchy) [13]: He points out that there are essential assumptions relating to infinitely small and large numbers these mathematicians must have made but never explicitly stated, which our own experience formalising Euler's work has confirmed.

2.3 Related Work

Over the years, researchers have spent much time analysing the works of historical mathematicians. Nowadays, interactive theorem-proving provides a highly appropriate tool for this type of proof analysis, requiring the user to examine minute details and explicitly deal with the rarest of edge cases to build a complete formal proof. It is no surprise then that there have been a number of similar projects formalising parts of different historical texts such as Hilbert's 'Grundlagen der Geometrie'[15, 3] or Newton's 'Principia Mathematica' [7].

Other parts of Euler's work have also been examined before, including an undergraduate project at the University of Edinburgh just this year [9], in which a different section of Euler's 'Foundations of Differential Calculus' dealing with the derivative of arcsine was formalised.

This is an active subject area and there is still much of historical mathematics to be formalised, so it is to be expected that the methods of interactive theorem proving will continue to be applied to assist in the analysis of historical works of mathematics.

Chapter 3

Technical Preliminaries

In this section, we will detail the reasoning for the two major decisions that needed to be made for this project: which mathematical framework and which proof assistant to use. We will also explain some of the notation used in the formalisation of nonstandard analysis in Isabelle and present the method Euler uses for differentiation.

3.1 Nonstandard Analysis

Euler based his work only on the existing notions of real numbers and his own musings about the infinite and infinitely small. Chapter 3 of his 'Foundations of Differential Calculus', entitled *On the Infinite and the Infinitely Small*, includes statements such as this:

> Let a be a finite quantity and let dx be infinitely small. Then $a+dx$ and $a-dx$, or, more generally, $a \pm n\,dx$, are equal to a. [5, pp.47-61]

Working only on the basis of the real numbers this seems very clearly wrong. However, what Euler means to say is that they are so close they may as well be equal, and Robinson's nonstandard analysis [18] provides a consistent formal framework in which we can explicitly represent this (and how it differs from equality) using the infinitely close relation.

Nonstandard analysis introduces the hyperreal numbers $^*\mathbb{R}$, an extension of the real numbers including infinitely large and infinitely small (infinitesimal) numbers. These can be defined as follows [10, ch. 3]:

ε is **infinitesimal** if $\quad |\varepsilon| < |r|$ for every $r \in \mathbb{R} \setminus \{0\}$

ω is **infinite** if $|\omega| > |r|$ for every $r \in \mathbb{R} \setminus \{0\}$

This is a fairly intuitive way of introducing the concepts of infinitesimals and infinitely large numbers. Formally, the hyperreal number field is constructed as equivalence classes on sequences of real numbers in the following way:

Let $\mathbb{R}^{\mathbb{N}}$ be the set of infinite sequences $r = \langle r_1, r_2, r_3, ...\rangle = \langle r_n : n \in \mathbb{N}\rangle$ of real numbers. The hyperreal numbers are then represented by equivalence classes on the elements of $\mathbb{R}^{\mathbb{N}}$ the following equivalence relation:

$$\langle r_n \rangle \equiv \langle s_n \rangle \qquad \text{iff} \qquad \{n \in \mathbb{N} : r_n = s_n\} \in \mathcal{F}$$

where $\mathcal{F}$ is some fixed nonprincipal ultrafilter on $\mathbb{N}$. The detailed definition of the ultrafilter construction can be found in [10], but for our purposes it suffices to say that $\mathcal{F}$ is a set of subsets of $\mathbb{N}$ and does not contain any sets of finite size. Intuitively, this means that $\langle r_n \rangle \equiv \langle s_n \rangle$ if they differ at only finitely many positions.

Let $[r] = \{s \in \mathbb{R}^{\mathbb{N}} : r \equiv s\}$ denote the equivalence class of the sequence $r \in \mathbb{R}^{\mathbb{N}}$. Each of these equivalence classes represents a hyperreal number. Every real number $x \in \mathbb{R}$ is represented by the equivalence class $[\mathbf{x}]$ the sequence $\langle x, x, x, ...\rangle$ belongs to. Addition and multiplication are defined entry-wise:

$$\begin{aligned} [r] + [s] &= [r \oplus s] &= [\langle r_n + s_n : n \in \mathbb{N}\rangle] \\ [r] \cdot [s] &= [r \odot s] &= [\langle r_s \cdot s_n : n \in \mathbb{N}\rangle] \end{aligned}$$

Following this definition, infinitesimals are the equivalence classes of sequences approaching zero, while infinitely large numbers are the equivalence classes of sequences that do not have an upper bound. It should be noted that this means 0 is an infinitesimal (and the only infinitesimal that is also a real number).

To compare hyperreal numbers, we can use the *infinitely close relation*: It expresses that aside from infinitesimal (infinitely small) differences, the real parts of two hyperreal numbers are equal. In contrast, equality of course still exists on hyperreal numbers, but requires them to be in the same equivalence class with respect to the relation above ($[r] = [s]$, which holds iff $\langle r_n \rangle \equiv \langle s_n \rangle$), meaning they only differ in a finite number of places. Two numbers a, b that are *infinitely close* (denoted $a \approx b$) may differ in more places, but their difference $a - b = c$ must be infinitesimal (which can also be expressed as $c \approx 0$).

A very important result Robinson presented with his nonstandard analysis is the *transfer principle*: It states that, for a certain subset of formal language, a statement holds for the real numbers if and only if it holds for the hyperreal numbers as well. The construction of this subset is detailed in [10, ch. 4]. It assigns every set, function and relation a **-transform*, which extends them to the hyperreal numbers. For example, the *-transform of a function f is $*f[\langle x_n \rangle] = [\langle f(x_n) : n \in \mathbb{N} \rangle]$. Applying the transfer principle to a rule on reals then replaces all occurrences of those by their *-transforms to turn it into a rule on hyperreals. This allows us to use many rules previously proven to hold for the real numbers without having to prove them anew for the hyperreals. In our case, it means we can assume the properties of the natural logarithm on reals since they can be transferred to apply to hyperreals as well.

Part of the underlying construction for the transfer principle are the concepts of *internal functions* and *internal sets*. An internal function f is a function on hyperreals that can be expressed as a sequence of functions f_n on the individual sequence components r_n of a hyperreal number $[r] = [\langle r_n : n \in \mathbb{N} \rangle]$ such that $f([r]) = [\langle f_n(r_n) : n \in \mathbb{N} \rangle]$. Similarly, an internal set A can be expressed as a sequence of sets A_n for every $n \in \mathbb{N}$.

3.2 Isabelle

Computer-assisted theorem proving is a growing field, mainly divided into two sections: *automated theorem proving*, where a computer tool is given a statement and seeks to build a proof for that statement without further input from the user, and *interactive theorem proving*, where the user can break a statement into smaller steps, specify which rules or proof techniques should be applied etc and the tool then verifies the correctness of that step using those rules. There are multiple reasons to use an interactive theorem prover rather than an automated one: While automated theorem provers have become incredibly powerful over the years, there are still roadblocks that keep them from being applicable for bigger projects - one such example is using induction as a proof technique, which quickly makes the search space to be examined by an automated theorem prover quite intractable [2]. But there is another major reason to use an interactive theorem prover for this project: We are not interested in just reproducing Euler's results by any means possible. As the goal of this project is to follow his reasoning and verify his arguments, it is important that we are able to build a proof that attempts to follow the same line of argument as Euler's original writing. An interactive theorem prover allows us to specify each proof step, which makes it particularly suitable for this purpose.

These days there are many different interactive theorem proving systems to choose from, each with their own approach, advantages and drawbacks. For this project we chose to work with Isabelle, for a number of reasons:

Isabelle [16] is a generic theorem prover, meaning it can be used with different object logics. The most commonly used one - and the one we are using for this project - is Higher-Order Logic (HOL), but there are a number of different environments for other object logics such as Zermelo-Fraenkel set theory. As one of the major theorem provers, Isabelle has a large established user base and is constantly being extended through contributions from all over the world. One notable extension was the addition of the more readable *Isar* proof style [20], which resembles a written mathematical proof and is therefore well suited to for projects such as this one. Isabelle also offers a great deal of automation both in proof methods and with its powerful proof search tool *sledgehammer* [17], which invokes a number of automated theorem provers on a given subgoal and, if they are successful in constructing a proof for that subgoal, attempts to use the information about rules used etc. to construct an Isabelle proof for that subgoal. This can save the user a lot of time and is therefore incredibly useful in proof development.

But one of the main arguments for using Isabelle for this project is that we are using nonstandard analysis, and Isabelle is one of the few theorem provers that offer an existing formalisation of nonstandard analysis [8] which we can build on. This is crucial since the time frame of this project would not allow for building a formalisation from scratch. Moreover, because there is an existing formalisation of nonstandard analysis to build on, other similar projects have also been done using Isabelle [9, 12], which makes it easier to integrate results and useful supporting lemmata from those projects.

An Isabelle proof for any statement begins with the keyword **lemma** or **theorem** followed by the name the statement will be referenced by, often specifying the variables the proof **fixes** and which facts it **assumes** before declaring what statement it **shows**. This is followed by the proof body. In the Isar proof style, a proof environment is opened with **proof** and closed with **qed**. The proof typically consists of some number of statements of the form

(**have** | **hence**) [*proposition*] **using** [*some facts / rules*] **by** [*method*] followed by the closing line

(**thus** | **show**) [*goal*] **using** [*some facts / rules*] **by** [*method*]

The facts and rules can be referring to outside facts, such as in our case the formalisation of nonstandard analysis, or to previous statements within the same proof. In the latter case it may be useful to give a proposition a name to reference it by later (such as **have** A : [*proposition*] ...). A referenced general rule can be instantiated to specific variables or terms by referencing it as *rulename*[*of x y z*].

Depending on the complexity of the proposition, different proof methods can be used. Given the necessary facts and rules, many propositions can be proven using the *simp* and *auto* methods, but some cases may require a more specialised (e.g. *linarith*) or powerful (e.g. *blast*, *smt*) method.

3.3 Nonstandard Analysis in Isabelle

Much of the need to explain the notation we use for nonstandard analysis in Isabelle stems from the fact that Isabelle uses simple types, meaning it doesn't allow subtyping. Because of this we often need to perform explicit type conversions and use sets to express the different properties hyperreal numbers can have.

We use sets mainly to mark whether a number is an *Infinitesimal*, *HInfinite* (infinitely large) and *HFinite* (the corresponding inverse set, including infinitesimals). All of these are subsets of numbers of type *hypreal*, which is an abbreviation for *real star* (or $^*\mathbb{R}$). In Isabelle the type of a value *a* is denoted as *a::type*.

For type conversions, the model provides a number of different functions. Note that *'a* and *'b* are type variables, meaning the functions defined using these are polymorphic functions. We mainly use the following functions:

star_of :: 'a ⇒ 'a star	can be used to turn anything of type *'a* into the corresponding object of type *'a star*, such as a real to the corresponding hyperreal number or a natural to the corresponding hypernatural

hypreal_of_hypnat :: hypnat ⇒ hypreal	takes a hypernatural number and returns the corresponding hyperreal
**f* :: ('a ⇒ 'b) ⇒ 'a star ⇒ 'b star*	converts a function to its *-transform

To improve readability, these explicit type conversions will generally be omitted in the explanations accompanying our proofs.

In some cases the nonstandard equivalent of a function has its own separate definition. Examples relevant to this project are *pow*, the nonstandard version of exponentiation, which raises a hyperreal number to some hypernatural power, and the *hypersetsum* construct. The *hypersetsum* $[\langle f_n \rangle]\ [\langle A_n \rangle] = [\langle \mathit{sum}\ f_n\ A_n : n \in \mathbb{N} \rangle]$ is the (finite) sum of the function f_n applied to every element of the set A_n at every sequence position n. To use this, $f = [\langle f_n \rangle]$ must be an internal function and $A = [\langle A_n \rangle]$ an internal set. For one of our Isabelle proofs in the next section, we use a *hypersetsum* over the set $\{0..N\}$ of hypernaturals from 0 to some (arbitrary but fixed) infinitely large N to model an infinite sum.

3.4 Modern Derivatives and Euler's Differentials

Nowadays, the commonly used approach to express the derivative of a function $f(x)$ is using limits [19]:

$$f'(x) = \lim_{h \to 0} \frac{f(a+h) - f(a)}{h}$$

As mentioned in the previous chapter, limits are a concept that did not exist during Euler's time, and in fact only came around as a more rigorous framework to express the ideas of Leibniz, Euler and their contemporaries.

It is not surprising then that the form Euler uses ends up looking very similar to the one above, with the exception of the use of limits. Euler uses an infinitely small number dx in place of the variable h. He also first specifies the 'differential' $dy = f(x+dx) - f(x)$ of a function $y = f(x)$, and spends most of his time detailing the ways to calculate the differentials of different variations of the function f - mainly, this means explaining how the differential is affected by products of functions or nested function applications. He then calculates what we would call the derivative of a function by working out the ratio of $\frac{dy}{dx}$:

$$\frac{dy}{dx} = \frac{f(x+dx) - f(x)}{dx}$$

It should be noted that he usually achieves this by calculating the value of the differential and dividing by dx on both sides. This poses a problem that we will encounter later: In our model the resulting statement about the value of the differential is not an equality but instead uses the infinitely close relation, which isn't necessarily preserved when dividing by an infinitesimal on both sides (since that is equivalent to multiplying by $\frac{1}{dx}$, which is infinitely large for infinitesimal dx).

He also goes into higher differentials and higher derivatives: He presents the second differential $d^2y = dy^I - dy = (f(x+2dx) - f(x+dx)) - (f(x+dx) - f(x))$ and goes on to list the third and fourth differential to imply a general rule for the nth differential. From there, he gets the nth derivative $\frac{d^ny}{dx^n}$, by dividing this by dx^n. Again, this causes some complications, as we will see in the following chapter.

Chapter 4

Isabelle Proofs

In this chapter, we will go over our Isabelle formalisations of the proofs on the differentiation of logarithms from chapter 6, *On the Differentiation of Transcendental Functions* [5, pp. 100-104]. The section has four major parts, extended by some examples Euler gives to support his reasoning. In a formal setting, those examples (given in paragraphs 182 & 185) are superfluous since the formal proof should be convincing enough, which is why we are focusing on the four parts containing proofs for the more general statements (paragraphs 180, 181, 183 and 184).

There are some common themes in all of the parts: Euler consistently makes some implicit assumptions, such as the arguments being > 0 since otherwise the logarithm is undefined. Where there are citations, they are not very specific, and often steps are not justified by a proof or a citation at all. Most results are expressed as equalities, but it is clear from Euler's explanations that he really means to say they are infinitely close. Some of the statements also have their own more specific issues, which we will detail in the following.

4.1 180 - Differential of $\ln x$

This section proves that if $y = \ln x$ then $dy = \frac{dx}{x}$. Euler starts the proof off by stating that if $y + dy = \ln(x + dx)$, then $dy = \ln(x + dx) - \ln x = \ln(1 + \frac{dx}{x})$. The last step here skips over a handful of transformations applying basic logarithm rules to get to the form $\ln(1 + \frac{dx}{x})$.

He then invokes an infinite sum definition of $\ln(1 + z)$ given in his 'Introduction to the Analysis of the Infinite' [4] which states that

$$\ln(1+z) = z - \frac{z^2}{2} + \frac{z^3}{3} - \frac{z^4}{4} + \dots$$

Substituting $\frac{dx}{x}$ for z yields

$$dy = \ln(1+\frac{dx}{x}) = \frac{dx}{x} - \frac{dx^2}{2x^2} + \frac{dx^3}{3x^3} - \dots$$

Finally, he states that "all of the terms of this series vanish in the presence of the first term" and concludes that $dy = \frac{dx}{x}$.

The proof structure given by Euler here is easy enough to follow. However, in addition to the relatively trivial transformations right at the start of the proof, there are a two other points in the proof where he doesn't provide an explicit proof for the step he is taking. The representation in nonstandard analysis also adds some complexity, which ends up making the Isabelle proof quite a bit more complicated than it looks on paper.

The first of these points is the infinite sum definition he references as an equivalent definition for $\ln(1+x)$. It is not immediately obvious where he takes this alternate definition from since there is only a footnote referencing a chapter in his 'Introduction to the Analysis of the Infinite' [4, ch. 7]. The infinite sum also needs to be represented slightly differently since we are using nonstandard analysis. We are modelling it using a *hypersetsum* construction. For the function f we use $\left(\lambda n.(-1)^n \cdot \frac{x^{(n+1))}}{n+1}\right)$, which applied to some natural number n yields the nth summand of our desired infinite sum. The set A we use is $\{0..N\}$, the set of all hypernatural numbers from 0 to some infinitely large N, which allows us to model summing to infinity as Euler does.

Since this alternate form follows from a previous proof of Euler's but is outside the scope of our project, we include it as an assumption, expressed as follows:

lemma *starfun-ln-powser*:
  assumes *N* ∈ *HNatInfinite*
  shows (*∗f∗ ln*) (*1* + *x*) ≈ *hypersetsum* (λ*n*. (−*1*) *pow n* ∗
      (*1* / *hypreal-of-hypnat*(*n* + *1*)) ∗ *x pow* (*n* + *1*)) {*0..N*}

We encounter the second complication when Euler declares that in the resulting infinite sum "all of the terms of this series vanish in the presence of the first term", which he does not provide any proof for at all. Given the *hypersetsum* definition above, this requires a proof that the sum from 0 can be separated into the first term and the sum from 1, and that the sum from 1 is infinitesimal. To separate the *hypersetsum* we needed to provide a proof for a side condition: that $(\lambda n.(-1)^n \cdot \frac{x^{(n+1)}}{n+1})$, the function we use in the *hypersetsum*, is internal. The proof for this fact (which will be referenced as *ln-powser-internalfun*) is included in appendix A.1.

There are a couple of different ways of going about proving that the sum from 1 is infinitesimal. The initial idea was to show that it fulfils the conditions of the *Infinitesimal* set, which we abandoned in favour of an inductive approach. However, for a direct inductive proof, an inductive principle like this one would have to be transferred from the real numbers:

lemma *hypersetsum-induct*:
$[\![\, F \in \mathit{InternalFuns}\ ;\ A \in \mathit{InternalSets}\ ;\ (\mathit{hypersetsum}\ \ F\ \{X..0\} \in A)\ ;$
$\forall n{::}\mathit{hypnat}.\ \mathit{hypersetsum}\ F\ \{X..n\} \in A \longrightarrow \mathit{hypersetsum}\ F\ \{X..\mathit{hSuc}\ n\} \in A\ ]\!]$
$\Longrightarrow (\forall n{::}\mathit{hypnat}.\ \mathit{hypersetsum}\ F\ \{X..n\} \in A)$

Even then, for this to be useful for this proof step there would have to be a definition of the set of Infinitesimals over the real numbers (an internal set definition), which cannot exist.

Ultimately, we found a much easier way to prove this: The natural logarithm function is continuous at $x > 0$. There is a definition of continuity in nonstandard analysis (called *isNSCont*) that uses infinitesimals, stating that if a function is continuous, then given two infinitely close inputs, the function outputs will also be infinitely close:

$[\![\mathit{isNSCont}\ ?f\ ?a;\ ?y \approx \mathit{star\text{-}of}\ ?a]\!] \Longrightarrow (*f*\ ?f)\ ?y \approx \mathit{star\text{-}of}\ (?f\ ?a)$

This means that an infinitesimal change in the input results in an infinitesimal change in the output, and so dy must be an infinitesimal. And since $\frac{dx}{x}$ is infinitesimal, it follows that the remainder of the sum must also be. The Isabelle proof for this statement (which will be referenced as *remainder-infinitesimal*) can be found in appendix A.2. [1]

Using these two results we can now provide a proof for $dy \approx \frac{dx}{x}$ with a structure similar to Euler's. Note the added assumptions about x being positive and nonzero (because the logarithm is undefined otherwise) and dx being infinitesimal and nonzero (to model Euler's intention). These are included in all of our proofs since they are necessary preconditions that Euler neglects to mention, but must have assumed.

[1] Since he does not provide any justification for this step, it is unclear whether Euler had this argument in mind or wanted to argue that the remainder of the sum is infinitely small without referring to dy. As Jacques Fleuriot was able to prove, the latter is also possible, albeit much more complicated.

theorem *dy-ln-x*:
 fixes *x*::*real* **and** *dx*::*hypreal*
 assumes $dx \in Infinitesimal - \{0\}$
  and $x > 0$
 shows (**f* ln*)((*star-of x*)+*dx*) − (**f* ln*)(*star-of x*) ≈ *dx* / (*star-of x*)
proof −
 have *XDX*:(*star-of x*) + *dx* > *0*
  using *assms Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
 obtain *dy* **where** *DY*: *dy* = (**f* ln*) (*star-of x* + *dx*) − (**f* ln*) (*star-of x*)
  by *auto*
 moreover have (**f* ln*)((*star-of x*) + *dx*) − (**f* ln*) (*star-of x*)
   = (**f* ln*) (((*star-of x*) + *dx*) / (*star-of x*))
  using *assms starfun-ln-div XDX* **by** *simp*
 moreover have (**f* ln*) (((*star-of x*) + *dx*) / (*star-of x*))
   = (**f* ln*) (*1* + *dx*/(*star-of x*))
  using *add-divide-distrib*[*of star-of x*] *assms div-self* **by** *simp*
 ultimately have *DY′*: *dy* = (**f* ln*) (*1* + *dx*/(*star-of x*))
  by *simp*

In order to be able to apply Euler's infinite sum definition, we use basic logarithm rules to transform $dy = \ln(x+dx) - \ln(x)$ to an expression including $\ln(1+z)$:

$$\begin{aligned} dy &= \ln(x+dx) - \ln(x) \\ &= \ln\left(\frac{x+dx}{x}\right) \\ &= \ln\left(1+\frac{dx}{x}\right) \end{aligned}$$

 obtain *N* **where** *N*: $N \in HNatInfinite$
  using *HNatInfinite-whn* **by** *blast*
 hence *DY″*:
  dy ≈ *hypersetsum* (λ*n*. ((−*1*) *pow n*) ∗ (*1*/*hypreal-of-hypnat*(*n* + *1*))
     ∗ ((*dx* / *star-of x*) *pow* (*n* + *1*))) {*0*..*N*}
  using *DY′ starfun-ln-powser* **by** *auto*

Now, after getting some (arbitrary but fixed) infinitely large hypernatural number N, we use our assumption about the existence of Euler's infinite sum form for $\ln(1+z)$, which we express using the *hypersetsum* construct.

 moreover have *HS*:
  hypersetsum (λ*n*. ((−*1*) *pow n*) ∗ (*1*/*hypreal-of-hypnat*(*n* + *1*))
     ∗ ((*dx* / *star-of x*) *pow* (*n* + *1*))) {*0*..*N*}
  = *hypersetsum* (λ*n*. ((−*1*) *pow n*) ∗ (*1*/*hypreal-of-hypnat*(*n* + *1*))
    ∗ ((*dx*/(*star-of x*)) *pow* (*n* + *1*))) {*1*..*N*} + (*dx* / *star-of x*)
 proof −
  have ((−*1*) *pow 0*)∗(*1*/*hypreal-of-hypnat*(*0* + *1*)) = *1*
   using *add.left-neutral div-by-1 hyperpow-minus-one2*[*of 0*]

```
    mult.right-neutral mult-zero-right of-hypnat-1 by auto
  hence dx/(star-of x) = ((−1) pow 0)∗(1/hypreal-of-hypnat(0 + 1))
                          ∗((dx/(star-of x)) pow (0 + 1))
   by auto
  hence dx/(star-of x) = hypersetsum (λn. (−1) pow n ∗
      (1/hypreal-of-hypnat(n + 1))∗((dx/(star-of x)) pow (n + 1))) {0}
   using ln-powser-internalfun assms(1) by auto
  moreover have hypersetsum (λn. (−1) pow n ∗
      (1/hypreal-of-hypnat(n + 1))∗ (dx/(star-of x)) pow (n + 1)) {0..N}
      =  hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
                          ∗ (dx/(star-of x)) pow (n + 1)) {0}
      +  hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
                      ∗ (dx/(star-of x)) pow (n + 1)) {hSuc 0..N}
   using hypersetsum-head-hSuc[of (λn. (−1) pow n ∗
     (1/hypreal-of-hypnat(n + 1))∗ (dx/(star-of x)) pow (n + 1)) 0 N]
     ln-powser-internalfun N assms(1) by auto
  moreover have hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
                      ∗ (dx/(star-of x)) pow (n + 1)) {hSuc 0..N}
         = hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
                        ∗ (dx/(star-of x)) pow (n + 1)) {1..N}
   using hSuc-def starfun-star-of[of Suc 0] by auto
  ultimately show hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
                     ∗ ( (dx / star-of x) pow (n + 1)) ) {0..N}
        = hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
          ∗ (dx/(star-of x)) pow (n + 1)) {1..N} + (dx / star-of x)
   by auto
 qed
```

The separation into the first term $\frac{dx}{x}$ and the remainder of the sum requires quite a bit of explicit manipulation of the *hypersetsum*, which is why this part of the proof is rather long. We need to prove that $\frac{dx}{x}$ is in fact the first term, then prove that the sum can be separated (both using the previously proven side condition that the function used is internal), and finally substitute $\frac{dx}{x}$ for the first term.

```
 ultimately have hypersetsum (λn. (−1) pow n ∗ (1/hypreal-of-hypnat(n + 1))
           ∗ (dx/(star-of x)) pow (n + 1)) {1..N} ∈ Infinitesimal
   using remainder-infinitesimal assms DY by auto
 hence hypersetsum (λn. (−1) pow n ∗(1/hypreal-of-hypnat(n + 1))
           ∗ ((dx/(star-of x)) pow (n + 1))) {1..N} + (dx / star-of x)
      ≈ dx / star-of x
   using Infinitesimal-add-approx-self2 approx-sym by blast
 thus ( ∗f∗ ln ) ((star-of x) + dx) − ( ∗f∗ ln ) (star-of x)
                                    ≈ dx / star-of x
   using DY DY'' HS approx-trans by auto
qed
```

For the final part of the proof, we plug in the proof that the remainder of the sum is infinitesimal, which allows us to prove that the whole sum is infinitely close to $\frac{dx}{x}$.

From that we can conclude that dy, or $\ln(x+dx)-\ln x$, is infinitely close to $\frac{dx}{x}$.

As we will see with many of these proofs, the general structure is the same as in Euler's version, but the formal proof is longer nevertheless. This is because, in a formal proof, there are often side conditions that must be explicitly proven which are not mentioned on paper, and larger proof steps may have to be broken down into smaller steps to guide the theorem prover through each individual change. In this case, we needed to provide proofs for two side conditions and take smaller steps while manipulating the *hypersetsum*.

4.2 181 - Differential of $\ln(px)$

The statement of this section is that for any function p of x, if $y=\ln p$ then $dy=\frac{dp}{p}$. Note that Euler uses p where we would use $p(x)$ and dp to refer to the differential $p(x+dx)-p(x)$.

Euler argues that this result follows directly from the previous one (paragraph 180) and can also be proven using an alternate form for the logarithm for which he again refers to his book 'Introduction to the Analysis of the Infinite' [4, ch. 7]. We could not identify which point in the chapter he is referring to since the alternate form he quotes is not mentioned anywhere as is. It is likely that he is referencing something derivable from some rule in the chapter, but it is not clear what that is.

Instead, we used the first method, deriving this result from paragraph 180 (referenced as *dy-ln-x*). However, this only works if the function p is continuous at x. Again, we use the nonstandard definition of continuity called *isNSCont*, assuming that *isNSCont* $p\,x$. It is safe to assume this, since Euler is working towards differentiation, and for a function to be differentiable at a given point it must be continuous at that same point.

Despite the argument being rather straightforward, the Isabelle proof for this statement is still 48 lines long. It can be found in appendix B. The first third introduces some abbreviations (mainly for readability) while making note of some facts that will be needed as side conditions later on. The main proof is split into two cases: $dp=0$ and $dp\neq 0$. If $dp=0$ then both sides of the resulting infinitely close statement are zero. If $dp\neq 0$, then we use *dy-ln-x* (the result of paragraph 180). This case split and the additional assumption of p being continuous at x are the only major differences between the proof on paper and in Isabelle.

4.3 183 - Products of Functions of x

This section considers three different configurations of products of functions inside a logarithm and provides proofs for their respective differentials: $\ln(p \cdot q \cdot r \cdot s)$, $\ln(\frac{p \cdot q}{r \cdot s})$ and $\ln(\frac{p^m \cdot q^n}{r^\mu \cdot s^\nu})$. Note that again, Euler uses p in place of $p(x)$ etc. Each of these statements is transformed and separated into sums of logarithms by basic logarithm rules, the differential of which can then be found using the results of the previous proofs on the summands. The Isabelle proofs for these are quite a bit longer than the proof on paper because they needed to be done in smaller explicit steps, but required no additional lemmas to be proven.

Since all three proofs are very similar, we will only present the proof for the first case, $ln(p \cdot q \cdot r \cdot s)$, in this section. The proofs for the other two cases are included in appendix C. For the first case, Euler states that given $y = \ln(p \cdot q \cdot r \cdot s)$, the resulting differential is $dy = \frac{dp}{p} + \frac{dq}{q} + \frac{dr}{r} + \frac{ds}{s}$. His justification for this is all of one line, arguing that if $y = \ln(p \cdot q \cdot r \cdot s)$ then $y = \ln p + \ln q + \ln r + \ln s$, from which he concludes that $dy = \frac{dp}{p} + \frac{dq}{q} + \frac{dr}{r} + \frac{ds}{s}$. Although he does not reference the previous paragraph explicitly, it is clear that he uses its result in the last step.

The Isabelle proof for this is, as one might expect, quite a bit longer than that:

```
theorem dy-ln-mult:
  fixes p q r s :: real and dp dq dr ds :: hypreal
 assumes dp ∈ Infinitesimal − {0} and dq ∈ Infinitesimal − {0}
   and dr ∈ Infinitesimal − {0} and ds ∈ Infinitesimal − {0}
   and p > 0 and q > 0 and r > 0 and s > 0
  shows ( ∗f∗ ln)((star-of p+dp)∗(star-of q+dq)∗(star-of r+dr)∗(star-of s+ds))
      − ( ∗f∗ ln) (star-of p ∗ star-of q ∗ star-of r ∗ star-of s)
     ≈ ( dp/star-of p )+( dq/star-of q )+( dr/star-of r )+( ds/star-of s )
proof −
 have P: star-of p + dp > 0 and Q: star-of q + dq > 0
  and R: star-of r + dr > 0 and S: star-of s + ds > 0
 proof−
  show P: star-of p + dp > 0
   using assms(1,5)Infinitesimal-add-hypreal-of-real-less by fastforce
  show Q: star-of q + dq > 0
   using assms(2,6)Infinitesimal-add-hypreal-of-real-less by fastforce
  show R: star-of r + dr > 0
   using assms(3,7)Infinitesimal-add-hypreal-of-real-less by fastforce
```

```
  show S: star-of s + ds > 0
    using assms(4,8)Infinitesimal-add-hypreal-of-real-less by fastforce
qed
have ( *f* ln)((star-of p+dp)*(star-of q+dq)*(star-of r+dr)*(star-of s+ds))
            − ( *f* ln) (star-of p*star-of q*star-of r*star-of s)
      = ((( *f* ln) (star-of p + dp) − ( *f* ln) (star-of p))
      +  (( *f* ln) (star-of q + dq) − ( *f* ln) (star-of q)))
      + ((( *f* ln) (star-of r + dr) − ( *f* ln) (star-of r))
      +  (( *f* ln) (star-of s + ds) − ( *f* ln) (star-of s)))
  using assms starfun-ln-mult P Q R S by auto
```

After proving the necessary side conditions, we use a basic property of the natural logarithm, the fact that $\ln(a \cdot b) = \ln a + \ln b$, to do the following transformation:

$$\begin{aligned} dy &= \ln(p+dp \cdot q+dq \cdot r+dr \cdot s+ds) - \ln(p \cdot q \cdot r \cdot s) \\ &= (\ln(p+dp)+\ln(q+dq)+\ln(r+dr)+\ln(s+ds)) - (\ln\ p+\ln\ q+\ln\ r+\ln\ s) \\ &= \ln(p+dp) - \ln\ p+\ln(q+dq) - \ln\ q+\ln(r+dr) - \ln\ r+\ln(s+ds) - \ln\ s \end{aligned}$$

In the proof above, these two steps are combined into one.

In his argument, Euler invokes the result of the previous section on p, q, r and s respectively to get $dy = \frac{dp}{p} + \frac{dq}{q} + \frac{dr}{r} + \frac{ds}{s}$. We can do the same in Isabelle:

```
moreover have  ((( *f* ln) (star-of p + dp) − ( *f* ln) (star-of p))
         +  (( *f* ln) (star-of q + dq) − ( *f* ln) (star-of q)))
         + ((( *f* ln) (star-of r + dr) − ( *f* ln) (star-of r))
         +  (( *f* ln) (star-of s + ds) − ( *f* ln) (star-of s)))
         ≈ dp/star-of p + dq/star-of q + dr/star-of r + ds/star-of s
proof −
  have ( *f* ln)(star-of p+dp) − ( *f* ln) (star-of p) ≈ dp/star-of p
    and ( *f* ln)(star-of q+dq) − ( *f* ln) (star-of q) ≈ dq/star-of q
    and ( *f* ln)(star-of r+dr) − ( *f* ln) (star-of r) ≈ dr/star-of r
    and ( *f* ln)(star-of s+ds) − ( *f* ln) (star-of s) ≈ ds/star-of s
    using dy-ln-x assms by auto
  hence (( *f* ln)(star-of p+dp) − (( *f* ln) (star-of p)))
      + (( *f* ln)(star-of q+dq) − (( *f* ln) (star-of q)))
       ≈ (dp/star-of p) + (dq/star-of q)
    and (( *f* ln)(star-of r+dr) − (( *f* ln) (star-of r)))
      + (( *f* ln)(star-of s+ds) − (( *f* ln) (star-of s)))
       ≈ (dr/star-of r) + (ds/star-of s)
    using approx-add by auto+
  hence  (( *f* ln) (star-of p + dp) − ( *f* ln) (star-of p))
     + (( *f* ln) (star-of q + dq) − ( *f* ln) (star-of q))
     + ((( *f* ln) (star-of r + dr) − ( *f* ln) (star-of r))
     +  (( *f* ln) (star-of s + ds) − ( *f* ln) (star-of s)))
     ≈ dp / star-of p + dq / star-of q + (dr / star-of r + ds / star-of s)
    using add.assoc approx-add by auto
```

thus *?thesis*
by (*simp add*: *semiring-normalization-rules*(25))
qed
ultimately show
$(*f* ln)((star\text{-}of\ p+dp)*(star\text{-}of\ q+dq)*(star\text{-}of\ r+dr)*(star\text{-}of\ s+ds))$
$- (*f* ln)\ (star\text{-}of\ p*star\text{-}of\ q*star\text{-}of\ r*star\text{-}of\ s)$
$\approx dp/star\text{-}of\ p + dq/star\text{-}of\ q + dr/star\text{-}of\ r + ds/star\text{-}of\ s$
by *auto*
qed

Here, after invoking *dy-ln-x* on p, q, r and s, we need to construct the resulting sum in smaller steps, explicitly providing the addition rule for the infinitely close relation. The proofs for the other two cases are included in appendix C. They follow the same general structure, just using some additional logarithm rules to account for division and exponentiation.

For all three of them, the line of reasoning we use is true to Euler's intention, although in this paragraph especially he does not prove his statements as much as suggest an idea of a proof. Consequently, the Isabelle proofs are much longer than the ones given in the book.

4.4 184 - Higher Differentials and Derivatives

In this section, Euler goes into higher differentials $d^n y$ (differentials of differentials). He illustrates the pattern they follow by explicitly stating the first four. He also specifies what $\frac{d^n y}{dx^n}$ evaluates to for each of them, specifying the derivative for the first time in the chapter. Although it is not stated in the text, he clearly means to imply a general rule, so we will focus on the proof for the general rule rather than just the first four instances. Since we want to prove this by induction, we start by formalising the first derivative.

4.4.1 First Derivative

The main challenge of this section is that Euler's argument to get the derivative from the differential involves taking the result of paragraph 180 and transforming that equality - which we are modelling using the infinitely close relation - by dividing by dx, an infinitesimal, on both sides. However, this does not necessarily preserve the infinitely close relation, so we need to find a different way to arrive at the same conclusion.

The same issue was encountered in a previous project [9], in which the section on the differentiation of arcsine was formalised. A similar workaround to the one proposed there also happens to work for our case: Instead of manipulating the infinitely close statement of the differential directly, we use the definition of the differential (an equality), transform it until we have the desired terms on either side multiplied by some factor, then prove that factor is infinitely close to 1.

Have $y = \ln x$. Then we can derive

$$
\begin{aligned}
& & dy &= \ln(x+dx) - \ln(x) \\
&\Leftrightarrow & dy &= \ln\frac{x+dx}{x} \\
&\Leftrightarrow & e^{dy} &= \frac{x+dx}{x} \\
&\Leftrightarrow & x+dx &= e^{dy}\cdot x \\
&\Leftrightarrow & dx &= (e^{dy}-1)\cdot x \\
&\Leftrightarrow & 1 &= \frac{(e^{dy}-1)\cdot x}{dx} \\
&\Leftrightarrow & 1 &= \frac{dy}{dy}\cdot\frac{(e^{dy}-1)\cdot x}{dx} \\
&\Leftrightarrow & 1 &= \frac{dy}{dx}\cdot\frac{(e^{dy}-1)\cdot x}{dy} \\
&\Leftrightarrow & \frac{1}{x} &= \frac{dy}{dx}\cdot\frac{(e^{dy}-1)\cdot x}{dy\cdot x} \\
&\Leftrightarrow & \frac{1}{x} &= \frac{dy}{dx}\cdot\frac{(e^{dy}-1)}{dy}
\end{aligned}
$$

Now we have $\frac{dy}{dx}$ on one side of the equation and the desired result, $\frac{1}{x}$, on the other. If we can prove that the factor $\frac{(e^{dy}-1)}{dy}$ is infinitely close to one then it follows that $\frac{dy}{dx} \approx \frac{1}{x}$.

The proof that the factor is infinitely close to 1 follows directly from the fact that the derivative of the exponential function at point 0 is 1: The nonstandard version of the derivative called *NSDERIV* is defined as follows:

NSDERIV ?f ?x :> *?D* =
(∀*h*∈*Infinitesimal* − {*0*}.
  ((∗*f*∗ *?f*) (*star-of ?x* + *h*) − *star-of* (*?f ?x*)) / *h* ≈ *star-of ?D*)

Thus since *NSDERIV exp* $0 :> 1$ we have $e^{0+h} - e^0 \approx 1$ for any infinitesimal $h \neq 0$, including dy. To use this we need to provide proofs for $dy \neq 0$ and $dy \in$ *Infinitesimal*, both of which can be found in appendix D. Note that in this section we use the function $dy\ f\ x\ dx$, which was defined by the previously referenced project [9]. It evaluates to $dy\ f\ x\ dx = f(x+dx) - f(x)$, which is exactly dy for $y = f\ x$. We use it here since it offers a shorthand for the term dy refers to that is easy to use in supplementary lemmas etc.

Using those facts, we can prove $\frac{(e^{dy}-1)}{dy} \approx 1$ in three short steps:

```
lemma factor-infcloseto-one:
  fixes x :: real and dx :: hypreal
  assumes dx ∈ Infinitesimal − {0} and x > 0
  shows ((( *f* exp) (dy ln x dx)) − 1) / (dy ln x dx) ≈ 1
proof −
  have NZ: dy ln x dx ≠ 0
    using dy-neq-zero assms by auto
  have NSDERIV exp 0 :> 1
    by (metis DERIV-exp NSDERIV-DERIV-iff exp-zero)
  thus (( *f* exp) (dy ln x dx) − 1) / (dy ln x dx) ≈ 1
    using nsderiv-def NZ NSDERIVD2 assms dy-infinitesimal by fastforce
qed
```

With this done, we are able to prove the base case / first derivative by formalising the equation transformation we gave at the beginning of the section and plugging in the result of the above lemma.

```
lemma firstderiv184:
  fixes x :: real and dx :: hypreal
  assumes dx ∈ Infinitesimal − {0} and x > 0
  shows (dy ln x dx) / dx ≈ 1 / (star-of x)
proof −
  have XDX: (star-of x) + dx > 0 using assms
    using Infinitesimal-add-hypreal-of-real-less by fastforce
  have dy ln x dx = ( *f* ln ) ((star-of x) + dx) − ( *f* ln ) (star-of x)
    by (simp add: seq-apply-function-def seq-delta-def)
  hence dy ln x dx = ( *f* ln) (((star-of x) + dx) / (star-of x))
    by (simp add: XDX add-pos-pos assms starfun-ln-div)
  hence ( *f* exp) (dy ln x dx) = ((star-of x) + dx) / (star-of x)
    using  assms starfun-exp-ln-iff[of ((star-of x) + dx)/ (star-of x)]
      XDX by auto
  hence (( *f* exp) (dy ln x dx)) ∗ (star-of x) = (star-of x) + dx
    using assms by auto
  hence (star-of x) + dx = (( *f* exp) (dy ln x dx)) ∗ (star-of x)
    by simp
  hence dx = ((( *f* exp) (dy ln x dx)) − 1) ∗ (star-of x)
    using left-diff-distrib'[of ( *f* exp) (dy ln x dx)] by auto
  hence dx = 1 ∗ ((( *f* exp) (dy ln x dx)) − 1) ∗ (star-of x)  by auto
  hence dx = ((dy ln x dx) / (dy ln x dx)) ∗
           ((( *f* exp) (dy ln x dx)) − 1) ∗ (star-of x)
```

using *dy-neq-zero[of dx x] assms div-self[of (dy ln x dx)]* **by** *auto*
hence *1 = (((dy ln x dx) / (dy ln x dx)) ∗*
(((∗f∗ exp) (dy ln x dx)) − 1) ∗ (star-of x)) / dx
using *assms* **by** *auto*
hence *1 = (dy ln x dx) / dx ∗*
((((∗f∗ exp) (dy ln x dx)) − 1) ∗ (star-of x) / (dy ln x dx))
by *auto*
hence *1 / (star-of x) = (dy ln x dx) / dx ∗*
((((∗f∗ exp)(dy ln x dx))−1)∗(star-of x)/(dy ln x dx))/star-of x
by *auto*
hence *1 / (star-of x) = (dy ln x dx) / dx ∗*
((((∗f∗ exp)(dy ln x dx))−1)∗(star-of x)/(dy ln x dx)/star-of x)
using *times-divide-eq-right[of (dy ln x dx)/dx*
(((∗f∗ exp)(dy ln x dx))−1)∗(star-of x)/(dy ln x dx) star-of x]
by *auto*
hence *A: 1 / (star-of x) = (dy ln x dx) / dx ∗*
((((∗f∗ exp)(dy ln x dx)) − 1) / (dy ln x dx))
using *assms(2)* **by** *auto*

Up to this point, we are able to formalise the transformations almost exactly as described at the beginning of this section. We only needed one additional intermediate step, introducing the factor 1 before substituting $\frac{dy}{dy}$ in the same place:

$$
\begin{aligned}
& & 1 &= \frac{(e^{dy}-1)\cdot x}{dx} \\
&\Leftrightarrow & 1 &= 1\cdot\frac{(e^{dy}-1)\cdot x}{dx} \\
&\Leftrightarrow & 1 &= \frac{dy}{dy}\cdot\frac{(e^{dy}-1)\cdot x}{dx}
\end{aligned}
$$

moreover have *(dy ln x dx) / dx ∗ ((∗f∗ exp)(dy ln x dx) − 1) / (dy ln x dx)*
≈ (dy ln x dx) / dx ∗ 1
proof −
have *approx1: ((((∗f∗ exp)(dy ln x dx)) − 1)/(dy ln x dx)) ≈ 1*
using *factor-infcloseto-one assms* **by** *auto*
moreover have *((dy ln x dx)/dx) ∈ HFinite*
proof *(rule ccontr)*
assume *dy ln x dx / dx ∉ HFinite*
hence *dy ln x dx / dx ∈ HInfinite*
using *HInfinite-HFinite-disj* **by** *blast*
moreover have *((∗f∗ exp)(dy ln x dx) − 1)/(dy ln x dx)*
∈ HFinite − Infinitesimal
using *HFinite-diff-Infinitesimal-approx[of - 1] approx1*
not-HInfinite-one one-not-Infinitesimal
HInfinite-diff-HFinite-Infinitesimal-disj **by** *blast*
ultimately have *1 / (star-of x) ∈ HInfinite*
using *HInfinite-HFinite-not-Infinitesimal-mult[of (dy ln x dx)/dx*
((((∗f∗ exp)(dy ln x dx)) − 1)/(dy ln x dx))] A **by** *auto*

moreover have *1 / (star-of x)* ∈ *HFinite*
using *HFinite-star-of[of 1/x]* **by** *auto*
ultimately show *False*
using *HFinite-HInfinite-iff* **by** *blast*
qed
ultimately show
dy ln x dx / dx ∗ *((* ∗*f*∗ *exp) (dy real-ln x dx) − 1) / dy real-ln x dx*
≈ *dy real-ln x dx / dx* ∗ *1*
using *approx-mult2[of ((* ∗*f*∗ *exp) (dy real-ln x dx) − 1)*
/ dy real-ln x dx 1 ((dy ln x dx)/dx)] **by** *auto*
qed
ultimately show *(dy ln x dx) / dx* ≈ *1 / star-of x*
using *approx-sym mult.comm-neutral* **by** *auto*
qed

The entire second half of the proof is spent arguing that, since the factor we identified is infinitely close to one, $\frac{dy}{dx}$ multiplied by that factor is infinitely close to $\frac{dy}{dx} * 1$. This is not immediately obvious since two infinitely close values $a \approx b$ need not be infinitely close anymore when both multiplied by some factor c: If the factor c is infinitely large, then we can't make a judgement about whether $a \cdot c \approx b \cdot c$. For this reason, we need $\frac{dy}{dx} \in HFinite$, which we prove by contradiction. Finally, from this and the result of the transformations above, it follows that $\frac{dy}{dx} \approx \frac{1}{x}$

Using this modified version of the workaround used in the formalisation of Euler's differentiation of arcsine [9] we were able to reproduce Euler's result for the first derivative of $\ln x$. The same proof could have been constructed at the time since the transformations only use rules that were also available to Euler.

4.4.2 Higher Derivatives

For the higher derivatives, Euler presents

$$\frac{dy}{dx} = \frac{1}{x} \qquad \frac{d^2y}{dx^2} = \frac{-1}{x^2}$$
$$\frac{d^3y}{dx^3} = \frac{2}{x^3} \qquad \frac{d^4y}{dx^4} = \frac{-6}{x^4}$$

From these examples we can identify the general form he is implying:

$$\frac{d^ny}{dx^n} = \frac{(-1)^{(n-1)} \cdot (n-1)!}{x^n}$$

Euler's argument for the higher derivatives is exactly the same as for the first, dividing the result of the corresponding higher differential $d^n y$ by dx^n to get the higher derivative, which means we are confronted with the same problem of dividing by an infinitesimal again.

However, it turns out we can actually prove the general statement for the higher derivatives without using higher differentials at all: The inductive proof we will detail in this section uses the first derivative statement we proved above as the base case, and the induction step mainly hinges on the fact that the derivative of $\frac{(-1)^n \cdot n!}{x^{(n+1)}}$ is $\frac{(-1)^{(n+1)} \cdot (n+1)!}{x^{(n+2)}}$, which we can prove using some basic derivative rules:

$$\begin{aligned}
\left(\frac{(-1)^n \cdot n!}{x^{(n+1)}}\right)' &= (-1)^n \cdot n! \cdot \left(\frac{1}{x^{(n+1)}}\right)' \\
&= (-1)^n \cdot n! \cdot \left(\frac{1}{z}\right)' \cdot \left(x^{(n+1)}\right)' && \text{(chain rule, } z = x^{(n+1)}\text{)} \\
&= (-1)^n \cdot n! \cdot \frac{-1}{z^2} \cdot \left(x^{(n+1)}\right)' \\
&= (-1)^n \cdot n! \cdot \frac{-1}{z^2} \cdot (n+1) \cdot x^n \\
&= (-1)^n \cdot n! \cdot \frac{-1}{(x^{(n+1)})^2} \cdot (n+1) \cdot x^n && \text{(substitute } z = x^{(n+1)}\text{)} \\
&= (-1)^n \cdot (-1) \cdot n! \cdot (n+1) \cdot \frac{x^n}{x^{(n+1)\cdot 2}} \\
&= (-1)^{(n+1)} \cdot (n+1)! \cdot \frac{x^n}{x^n \cdot x^{(n+2)}} \\
&= \frac{(-1)^{(n+1)} \cdot (n+1)!}{x^{(n+2)}}
\end{aligned}$$

Euler covers all of the necessary rules except for the chain rule in chapter 5, *On the Differentiation of Algebraic Functions of One Variable* [5, pp.77-98]. He never generalises enough to get to the chain rule, but its earliest appearance in print is in L'Hospital's 'Analyse des infiniment petits' [14], which was published in 1696, over 50 years before the publication of Euler's 'Foundations of Differential Calculus' in 1755. Thus this entire calculation could have been done with the knowledge available during Euler's time.

To prove this via induction, we ended up using a recursive definition of the higher derivative which was also used in the previously referenced project [9]:

fun *NSDERIV-higher* :: [*nat*, *real* ⇒ *real*, *real*, *real* ⇒ *real*] ⇒ *bool*

```
  where
   nsderiv-higher-zero: NSDERIV-higher 0 f x d ⟷ f = d |
   nsderiv-higher-Suc: NSDERIV-higher (Suc n) f x d ⟷
                (∃D. NSDERIV-higher n f x D ∧ NSDERIV D x :> d x)
```

Using this, the Isabelle proof for the general statement works as follows:

```
theorem higherderiv184-general:
  fixes n::nat and x::real
  assumes x > 0 and n ≥ 0
  shows NSDERIV-higher (n+1) ln x (λ x.((−1)^n ∗ (fact n)) / (x ^ (n+1)))
proof (induct n)
  case 0
  moreover have NSDERIV ln x :> (λa. (− 1)^0 ∗ fact 0 / a^1) x
    unfolding nsderiv-def
  proof
    fix h::hypreal
    assume H: h ∈ Infinitesimal − {0}
    moreover have star-of ((− 1) ^ 0 ∗ fact 0 / x ^ 1) = 1 / (star-of x)
      using hyperpow-hypnat-of-nat[of −1 0] by auto
    moreover have dy ln x h / h ≈ 1 / star-of x
      using firstderiv184 H assms by auto
    ultimately show (( ∗f∗ ln) (star-of x + h) − star-of (ln x)) / h
                 ≈ star-of ((− 1) ^ 0 ∗ fact 0 / x ^ 1)
      by (simp add: seq-apply-function-def seq-delta-def)
  qed
  thus ?case by simp
```

The base case follows almost immediately from the result of the previous subsection (referenced as *firstderiv184*). The only additional step we need is to show that that result matches the general form at $n = 0$. Note that we use $(n+1)$ in the goal because we want to show this is true from the first derivative upwards, and natural induction in Isabelle starts at 0.

```
next
  case (Suc n)
  have NSDERIV (λx. (− 1) ^ n ∗ fact n / x ^ (n + 1)) x :>
          (λx. (− 1) ^ Suc n ∗ fact (Suc n) / x ^ (Suc n + 1)) x
  proof −
    have A: NSDERIV (λx. 1 / x ^ (n + 1)) x :> (−1)∗(Suc n) / x^(Suc n + 1)
    proof −
      have neq-0:x^(n+1) ≠ 0 using assms by auto
      moreover have (λx. 1 / x ^ (n + 1)) = (λx. 1/ x) ∘ (λy. y^(n+1))
        by auto
      moreover have NSDERIV ((λx. 1 / x) ∘ (λy. y^(n+1))) x :>
                              (−1)∗(Suc n) / x^(Suc n + 1)
      proof −
        have NSDERIV (λy. y^(n+1)) x :> (n+1) ∗ x^(n)
```

```
    using NSDERIV-pow[of n+1] by auto
  moreover have NSDERIV ( λx. 1/ x) ((λy. yˆ(n+1)) x) :>
                    (−1) / (x ˆ (n + 1)) ˆ 2
  proof −
    have NSDERIV ( λx. 1/ x) ((λy. yˆ(n+1)) x) :>
          (0 ∗ x ˆ (n + 1) − 1 ∗ 1) / (x ˆ (n + 1)) ˆ Suc (Suc 0)
      using NSDERIV-quotient[of λx. 1
        ((λy. yˆ(n+1)) x) 0 ( λx. x)] neq-0 by auto
    thus ?thesis
      by (simp add: numeral-2-eq-2)
  qed
  ultimately have NSDERIV ((λx. 1/ x) ∘ (λy. yˆ(n+1))) x :>
          (−1) / (x ˆ (n + 1)) ˆ 2 ∗ ( (n + 1) ∗ x ˆ n)
    using NSDERIV-chain[of λx. 1/x (λy. yˆ(n+1)) x
      (−1) / (x ˆ (n + 1)) ˆ 2 ] by auto
  moreover have (−1) / (x ˆ (n + 1)) ˆ 2 ∗ ( (n + 1) ∗ x ˆ n)
                    = (−1)∗(Suc n) / xˆ(Suc n + 1)
  proof −
    have (−1) / (x ˆ (n + 1)) ˆ 2 ∗ (real (n + 1) ∗ x ˆ n)
        = (−1)∗ x ˆ n / (x ˆ (n + 1)) ˆ 2 ∗  (n + 1) by auto
    hence (−1) / (x ˆ (n + 1)) ˆ 2 ∗ (real (n + 1) ∗ x ˆ n)
        = (−1)∗ x ˆ n / (x ˆ n ∗ xˆ(n + 2)) ∗  (n + 1)
      by (simp add: numeral-2-eq-2)
    hence (−1) / (x ˆ (n + 1)) ˆ 2 ∗ (real (n + 1) ∗ x ˆ n)
        = (−1) ∗ (1 / xˆ(n + 2)) ∗ (n + 1)
      by simp
    hence (−1) / (x ˆ (n + 1)) ˆ 2 ∗ (real (n + 1) ∗ x ˆ n)
        = (−1) ∗ (n + 1) ∗ (1 /  xˆ(n + 2))
      using mult.commute[of  (1 / xˆ(n + 2)) n+1] by linarith
    thus ?thesis by auto
  qed
  ultimately show ?thesis by auto
 qed
 ultimately show ?thesis
  by smt
qed
```

For the induction step, we use almost exactly the chain of reasoning detailed above. We isolate the constant factor $(-1)^n \cdot n!$ and prove that the derivative of $\frac{1}{x^{(n+1)}}$ is $\frac{(-1)\cdot(n+1)}{x^{(n+2)}}$, first applying the various derivation rules and then, in the last third of this part, rearranging the resulting term.

```
thus NSDERIV (λx. (− 1) ˆ n ∗ fact n / x ˆ (n + 1)) x :>
    (− 1) ˆ Suc n ∗ fact (Suc n) / xˆ(Suc n + 1)
proof −
  have B: NSDERIV (λx. (− 1) ˆ n ∗ fact n ∗ (1 / x ˆ (n + 1))) x :>
      (− 1) ˆ n ∗ fact n ∗  (− 1 ∗ (Suc n)) / x ˆ (Suc n + 1)
    using NSDERIV-cmult[of  (λx. 1/ x ˆ (n + 1)) -
```

```
        (−1)∗(Suc n) / x^(Suc n + 1) (− 1) ^ n ∗ fact n] A by auto
      have (− 1) ^ n ∗ fact n ∗  (− 1 ∗ (Suc n))
           = (− 1) ^ n ∗(− 1) ∗ fact n ∗ (Suc n)
        by linarith
      hence (− 1) ^ n ∗ fact n ∗  (− 1 ∗ (Suc n))
           = (− 1) ^ (Suc n) ∗ fact (Suc n) by auto
      hence (− 1) ^ n ∗ fact n ∗  (− 1 ∗ (Suc n)) / x ^ (Suc n + 1)
         = (− 1) ^ Suc n ∗ fact (Suc n) / x^(Suc n + 1) by auto
      thus ?thesis using B by auto
    qed
  qed
  moreover from Suc have NSDERIV-higher (n + 1) ln x
                        (λx. (− 1) ^ n ∗ fact n / x ^ (n + 1))
    by simp
  ultimately show NSDERIV-higher (Suc n + 1) ln x
                (λx. (− 1) ^ Suc n ∗ fact (Suc n) / x ^ (Suc n + 1))
    by auto
qed
```

Finally, we reintroduce the factor $(-1)^n \cdot n!$ and rearrange the resulting term to complete our proof that the derivative of $\frac{(-1)^n \cdot n!}{x^{(n+1)}}$ is $\frac{(-1)^{(n+1)} \cdot (n+1)!}{x^{(n+2)}}$. Using this and the induction hypothesis (referenced as *Suc*) we can prove our goal, since by definition $\mathit{NSDERIV\text{-}higher}(\mathit{Suc}\ n)\ f\ x\ d \longleftrightarrow (\exists D.\ \mathit{NSDERIV\text{-}higher}\ n\ f\ x\ D \wedge \mathit{NSDERIV}\ D\ x\ :> d\ x)$.

Again, this is quite different from the proof Euler presents in the book. However, as previously mentioned, all the rules used in this proof were known at the time, so Euler could have argued the same way.

4.4.3 Equivalence of Higher Derivative Definitions

To tie this back to the $\frac{d^n y}{dx^n}$ form Euler uses, it would be useful to prove that the two forms are equivalent. However, this turns out to be more problematic than we expected, mainly due to the number of nested occurrences of the infinitely close relation, leading to a similar issue we had to circumvent for the proof of the derivatives of ln x: having an infinitely close statement which needs to be divided by an infinitesimal on both sides. Unlike in that case, since this would have to be a general proof for any function f and not just specifically for ln, we cannot use the same workaround used for the base case of the above proof, which relies on the natural logarithm having an inverse and a distributive rule with subtraction.

Unfortunately, we were not able to complete a proof for this equivalence in the time available for this project. However, we will detail the most promising attempt and its complications in this section. We hope that this may be helpful for future similar projects who will likely encounter this same issue.

To express higher differentials according to Euler's definitions, we defined a function *diff a b f x dx*, which calculates the ath differential of the function f at position x. The role of b is to specify which sequence entry is given to f as an argument: When introducing higher differentials, Euler also introduces the notation $y = f(x)$, $y^I = f(x+dx)$, $y^{II} = f(x+2dx)$ etc. He then defines $dy = y^I - y$, $dy^I = y^{II} - y^I$ etc. and $d^2y = dy^I - dy$, $d^3y = d^2y^I - d^2y$ etc. We can sum this up in the form of three rules:

$$y^b = f(x+(b\cdot dx))$$
$$dy^b = y^{(b+1)} - y^b$$
$$d^a y^b = d^{(a-1)}y^{(b+1)} - d^{(a-1)}y^b$$

The function *diff a b f x dx* calculates $d^a y^b$ and is defined in Isabelle as:

```
fun diff :: nat ⇒ nat ⇒ (hypreal ⇒ hypreal) ⇒ hypreal ⇒ hypreal ⇒ hypreal
 where
  diff-zero: diff 0 b f x dx = f (x + star-of(b)∗dx) |
  diff-Suc: diff (Suc a) b f x dx = diff a (b+1) f x dx − diff a b f x dx
```

We defined this on hyperreals instead of reals to be able to express the following property, which we anticipated would be useful for the proof:

```
lemma diff-seq:
 diff a b f (x + dx) dx = diff a (b+1) f x dx
proof (induct a arbitrary: b)
 case 0
 have diff 0 b f (x + dx) dx = f ((x + dx) + star-of(b)∗dx)
  by auto
 moreover have A:... = f (x + (dx + star-of(b)∗dx))
  by (simp add: semiring-normalization-rules(21))
 moreover have ... = f (x + (star-of b + 1)∗dx)
  by (simp add: semiring-normalization-rules(3))
 ultimately show ?case
  by (metis diff-zero
    of-nat-1 of-nat-add star-of-of-nat)
next
 case (Suc a)
 have diff (Suc a) b f (x + dx) dx
```

$= diff\ a\ (b+1)\ f\ (x+dx)\ dx - diff\ a\ b\ f\ (x+dx)\ dx$
by *auto*
moreover have ... $= diff\ a\ (b+2)\ f\ x\ dx - diff\ a\ (b+1)\ f\ x\ dx$
using *Suc* **by** *auto*
ultimately show *?case*
by *simp*
qed

Using this definition to express the higher differential $d^n y$, the equivalence statement we want to prove is the following:

lemma *nsderiv-higher-equiv*:
fixes *x*::*real*
assumes $x > 0$
shows $NSDERIV\text{-}higher\ (n{+}1)\ f\ x\ d \longleftrightarrow (\forall\ h \in Infinitesimal - \{0\}.$
$star\text{-}of\ (d\ x) \approx (diff\ (n{+}1)\ 0\ (\ {*}f{*}\ f)\ (star\text{-}of\ x)\ h)$
$/\ (h\ pow\ star\text{-}of\ (n{+}1)))$

Our approach is to prove this by induction. Again, we use $(n+1)$ in the goal because we want to prove this statement for all $n \geq 1$, and in Isabelle natural induction starts at 0. The base case is fairly straightforward:

case *0*
have $NSDERIV\text{-}higher\ 1\ f\ x\ d = NSDERIV\ f\ x :> d\ x$
by *simp*
hence $NSDERIV\text{-}higher\ 1\ f\ x\ d = (\forall h{\in}Infinitesimal - \{0\}.$
$((\ {*}f{*}\ f)\ (star\text{-}of\ x + h) - star\text{-}of\ (f\ x))\ /\ h \approx star\text{-}of\ (d\ x))$
unfolding *nsderiv-def* **by** *simp*
moreover have $\forall h{\in}Infinitesimal - \{0\}.$
$(diff\ 1\ 0\ (\ {*}f{*}\ f)\ (star\text{-}of\ x)\ h)\ /\ (h\ pow\ star\text{-}of\ 1)$
$= ((\ {*}f{*}\ f)\ (star\text{-}of\ x + h) - star\text{-}of\ (f\ x))\ /\ h$
by *auto*
ultimately have $NSDERIV\text{-}higher\ 1\ f\ x\ d = (\forall h{\in}Infinitesimal - \{0\}.$
$star\text{-}of\ (d\ x) \approx (diff\ 1\ 0\ (\ {*}f{*}\ f)\ (star\text{-}of\ x)\ h)\ /\ (h\ pow\ star\text{-}of\ 1))$
using *approx-sym* **by** *auto*
thus *?case* **by** *auto*

Here, we only need to show that our goal at $n = 0$ matches the definition for the first derivative.

case *(Suc n)*
have $NSDERIV\text{-}higher\ (Suc\ n + 1)\ f\ x\ d \Longrightarrow$
$(\forall h{\in}Infinitesimal - \{0\}.\ hypreal\text{-}of\text{-}real\ (d\ x) \approx$
$diff\ (Suc\ n + 1)\ 0\ (\ {*}f{*}\ f)\ (hypreal\text{-}of\text{-}real\ x)\ h$
$/\ h\ pow\ hypnat\text{-}of\text{-}nat\ (Suc\ n + 1))$
proof –
assume $NSDERIV\text{-}higher\ (Suc\ n + 1)\ f\ x\ d$

then obtain *D* **where** *D1*:*NSDERIV-higher* $(n + 1)$ *f x D*
 and *D2*: *NSDERIV D x* :> *d x* **by** *auto*
hence *IH*: $\forall h \in$ *Infinitesimal* $- \{0\}$. *star-of* $(D\,x) \approx$
 diff $(n + 1)$ *0* $(*f* f)$ (*star-of x*) *h* / *h pow star-of* $(n + 1)$
 using *Suc* **by** *auto*
have *D2'*: $\forall$ *h* $\in$ *Infinitesimal* $- \{0\}$.
 $((*f* D)$ (*star-of x* + *h*) $-$ *star-of* (*D x*)) / *h* $\approx$ *star-of* (*d x*)
 using *D2* **unfolding** *nsderiv-def* **by** *auto*

For the induction step, we split the goal into two implications and attempted to prove the forward direction first. By definition *NSDERIV-higher (Suc n + 1) f x d* can be broken down to say there is some D that is the $(n+1)$th derivative at x and the derivative of Dx is dx.

To improve readability, we omit all occurences of $*f*$ and *star-of* in the following explanations and express *h pow n* using the notation for its standard equivalent, h^n.

At this point we would like to use the induction hypothesis (statement *IH*) to substitute $(D\,x)$ in $D2'$. However, because *IH* uses the infinitely close relation and the term we want to substitute for is divided by an infinitesimal in $D2'$, we cannot just substitute this and assume it preserves the infinitely close relation. $D\,(x+h)$ is even more problematic. Unfortunately, the definition of *NSDERIV-higher* only provides us with a function D that matches the $(n+1)$th derivative exactly at point x, but we do not know whether it is also the derivative at any other point, such as $(x+h)$. This is also hard to express in Isabelle since *NSDERIV* takes a function on reals f and a real number x as inputs, and thus we cannot use it to express the derivative at point $(x+h)$, which is hyperreal. However, one can argue that infinitesimal changes in the input should only cause infinitesimal changes in the output, and thus

$$
\begin{aligned}
& D\,(x+h) \\
\approx\; & (D\,x) && (1) \\
\overset{IH}{\approx}\; & \frac{\mathit{diff}\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}} \\
\approx\; & \frac{\mathit{diff}\,(n+1)\;0\;f\;(x+h)\;h}{h^{(n+1)}} && (2)
\end{aligned}
$$

We get (1) directly from the fact that D must be continuous at x to be differentiable. The proof for (2) is unfortunately much more involved (again due to the division by an infinitesimal) and was skipped due to time constraints.

Now assuming *IH* and the corresponding statement for $D(x+h)$, the way to resolve the division by an infinitesimal is to reformulate them into equalities. We consider two different approaches for this: The standard part and a factor infinitely close to one.

The standard part of a hyperreal number refers to the real number that hyperreal is infinitely close to. It exists for any finite hyperreal, which is any hyperreal that is not infinitely large, including infinitesimals. Thus, to transform an infinitely close statement to an equality, one must simply take the standard part on both sides, given that the terms are $\in HFinite$. And since $D\,x$ is a real, $st(D\,x) = D\,x$. The same does not hold for $D(x+h)$, but that is not the only problem with this approach. Even if we could substitute both to get

$$\begin{aligned} d(x) &\approx \frac{st\left(\frac{diff\ (n+1)\ 0\ f\ (x+h)\ h}{h^{(n+1)}}\right) - st\left(\frac{diff\ (n+1)\ 0\ f\ x\ h}{h^{(n+1)}}\right)}{h} \\ &= \frac{st\left(\frac{diff\ (n+1)\ 0\ f\ (x+h)\ h}{h^{(n+1)}} - \frac{diff\ (n+1)\ 0\ f\ x\ h}{h^{(n+1)}}\right)}{h} \end{aligned}$$

we cannot resolve this any further, since the division rule $st(A/B) = st(A)/st(B)$ requires $st(B) \neq 0$. But since h is an infinitesimal, $st(h) = 0$.

In an effort to find an alternative path, we proved that for any infinitely close statement on non-infinitesimals, there is some factor ≈ 1 which can transform it to an equality:

lemma *approx-factor*:
  fixes $A\ B$ :: *hypreal*
  assumes $A \approx B$ **and** $A \in HFinite$ **and** $B \in HFinite$
    and $A \notin Infinitesimal \vee B \notin Infinitesimal$
  shows $\exists\ a.\ a \approx 1 \wedge a * A = B$

The proof for this rule can be found in appendix D.3.

Using this, we get

$$d(x) \approx \frac{a \cdot \left(\frac{diff\ (n+1)\ 0\ f\ (x+h)\ h}{h^{(n+1)}}\right) - b \cdot \left(\frac{diff\ (n+1)\ 0\ f\ x\ h}{h^{(n+1)}}\right)}{h}$$

To resolve this further, we can express b in terms of a: Since both are ≈ 1, there is some infinitesimal c such that $b = a + c$. Substituting this, we can derive

$$d(x) \approx \frac{a \cdot \left(\frac{diff\ (n+1)\ 0\ f\ (x+h)\ h}{h^{(n+1)}}\right) - (a+c) \cdot \left(\frac{diff\ (n+1)\ 0\ f\ x\ h}{h^{(n+1)}}\right)}{h}$$

$$= \frac{a \cdot \left(\frac{diff\,(n+1)\;0\;f\;(x+h)\;h}{h^{(n+1)}} - \frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}}\right) - c \cdot \left(\frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}}\right)}{h}$$

$$= a \cdot \frac{diff\,(n+1)\;0\;f\;(x+h)\;h\; - diff\,(n+1)\;0\;f\;x\;h}{h^{(Suc\;n+1)}} - \frac{c \cdot \left(\frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}}\right)}{h}$$

This is where our lemma *diff-seq* comes in handy, as it allows us to transform $diff\,(n+1)\;0\;f\;(x+h)\;h$ to $diff\,(n+1)\;1\;f\;x\;h$. Using the definition of *diff*, we then get

$$d(x) \approx a \cdot \frac{diff\,(n+1)\;1\;f\;x\;h - diff\,(n+1)\;0\;f\;x\;h}{h^{(Suc\;n+1)}} - \frac{c \cdot \left(\frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}}\right)}{h}$$

$$= \quad a \cdot \frac{diff\,(Suc\;n+1)\;1\;f\;x\;h}{h^{(Suc\;n+1)}} - \frac{c}{h} \cdot \left(\frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}}\right)$$

The first term here is exactly what we want: the higher derivative statement in Euler's form for $(Suc\;n+1)$. The factor a is not an issue since we know it is ≈ 1. However, we are left with the second term, which we cannot prove to be infinitesimal. We know that $\frac{diff\,(n+1)\;0\;f\;x\;h}{h^{(n+1)}} \approx D\;x$ which is some finite real, so if $\frac{c}{h}$ is infinitesimal so is the whole term, but we do not have enough information about the ratio of c to h to prove this.

We were not able to resolve this problem or find a different way to prove this equivalence in the time available for this project. However, we hope this documentation of our approach will be useful for any future project tackling the same issue.

Chapter 5

Conclusion

We were able to provide proofs for all four main results presented in the section. As expected, the formal proofs were often much longer than their counterparts on paper. This has a couple of reasons which we anticipated: Formalisations of a proof in a theorem prover being roughly four times as long as the original proof is a known phenomenon referred to as the *De Bruijn factor* [21], which is certainly at play here. However, specifically in this project, there are also many points where Euler neglects to mention essential proof steps or side conditions, and other situations where what was an easy step using his model of infinitely small and large numbers is more complicated given the more rigorous framework of nonstandard analysis.

Most interesting to us was how closely we could follow Euler's exact reasoning. Paragraph 180 had some complications, but still generally followed the same structure as the proof on paper. The issues we encountered here were mainly due to missing proof steps and side conditions necessary because of the use of nonstandard analysis. The proof for paragraph 181 is also true to Euler's intention, generalising the argument of paragraph 180 to any function of x, with only one additional assumption. Paragraph 183 was the most straightforward to formalise since it consisted of mostly manipulations using the basic arithmetic rules for logarithms. However, to prove the results of paragraph 184, we had to diverge from Euler's way quite drastically. We ended up having to find an alternative path to what Euler does in the book, although we made sure to find one that could have been done only with the methods and rules available at the time.

The issue we encountered most often throughout the proofs was also the most straightforward to fix: Euler consistently neglects to mention preconditions for his proofs, specifically the arguments of the logarithm needing to be greater than zero. Since there is little use in trying to prove anything about the derivative of a function at a point where it is undefined, we resolved this by adding $x > 0$ as an assumption. Another added precondition is the continuity of the function p in the proof for paragraph 181. Euler never mentions it, but it turns out to be necessary for the proof. It is an unproblematic assumption though since we are working towards differentiation and if a function is differentiable at a point x, then it must also be continuous at x.

The biggest complication we had to deal with was in paragraph 184, where Euler calculates the derivative (and higher derivatives) by just taking his results about the differentials and dividing by dx (or for the higher derivatives dx^n) on both sides. As we explained, at this point in our proof the result we are working with is not an equation but an infinitely close statement, which is not necessarily preserved when both sides are divided by an infinitesimal. Since there is no direct way to circumvent this issue just at this step while preserving Euler's reasoning up until and thereafter, we had to take a different path. However, we were able to use a modified version of a workaround presented in a similar project [9] to solve this problem. This allowed us to prove the base case for the general statement about higher derivatives, and the induction step mainly involved applying known rules about the derivatives of simpler functions.

The fact that we were able to adapt a workaround used in another project [9] for the derivative suggests that this method may be generalisable to other cases as well. It varies slightly depending on the function used and requires the function to be invertible and to have a distributive law with subtraction. Because of those restrictions, it may be tricky to formalise a general rule using it, but this method will likely be applicable for other cases that fulfil those criteria.

For this project, we only examined one part of one chapter of Euler's 'Foundations of Differential Calculus'. There is still much of his work left to formalise, including the immediately following section dealing with the differentiation of exponentials.

An important component relevant to this work as well as to the other project [9] - and one that we were unable to complete due to time constraints - is proving equivalence between the way Euler expresses higher derivatives using higher differentials and the recursive definition *NSDERIV-higher*. We found this to be much more complicated than we anticipated, mostly due to necessary transformations involving the infinitely close relation in places where we divide by an infinitesimal. Proving this equivalence would be crucial for future formalisations of Euler's differential calculus, and we hope that the detailed account of our proof attempt may be useful towards that end.

Bibliography

[1] Jacques Bair, Piotr Błaszczyk, Robert Ely, Valérie Henry, Vladimir Kanovei, Karin U Katz, Mikhail G Katz, Semen S Kutateladze, Thomas McGaffey, David M Schaps, et al. Is mathematical history written by the victors? *Notices of the AMS*, 60(7):886–904, 2013.

[2] Alan Bundy. The Automation of Proof by Mathematical Induction. In *Handbook of Automated Reasoning*, pages 845–911. Elsevier, 2001.

[3] Christophe Dehlinger, Jean-François Dufourd, and Pascal Schreck. Higher-order intuitionistic formalization and proofs in Hilberts elementary geometry. In *International Workshop on Automated Deduction in Geometry*, pages 306–323. Springer, 2000.

[4] Leonhard Euler. *Introduction to Analysis of the Infinite*. Springer Science & Business Media, 2012.

[5] Leonhard Euler, John D Blanton, et al. *Foundations of differential calculus*. Springer New York, 2000.

[6] Howard Eves and Jamie H Eves. *An introduction to the history of mathematics*, volume 6. Holt, Rinehart and Winston New York, NY, 1964.

[7] Jacques Fleuriot. *A combination of geometry theorem proving and nonstandard analysis with application to Newtons Principia*. Springer Science & Business Media, 2012.

[8] Jacques D. Fleuriot and Lawrence C. Paulson. Mechanizing Nonstandard Real Analysis. *LMS Journal of Computation and Mathematics*, 3:140190, 2000.

[9] Zuzana Frankovska. *Exploring Eulers Foundations of Differential Calculus in Isabelle/HOL using nonstandard analysis*. 4th year project report, The University of Edinburgh, 2019.

[10] Robert Goldblatt. *Lectures on the hyperreals: an introduction to nonstandard analysis*, volume 188. Springer Science & Business Media, 2012.

[11] Evgenii Izrailevich Gordon, Anatoly G Kusraev, and Semen Samsonovich Kutateladze. *Infinitesimal analysis*, volume 544. Springer Science & Business Media, 2013.

[12] Kyriakos Katsamaktsis. *L'Hospitals Theorems and Eulers Notions of Orders of Infinity in Isabelle/HOL*. 4th year project report, The University of Edinburgh, 2018.

[13] D. Laugwitz. Hidden lemmas in the early history of infinite series. *aequationes mathematicae*, 34(2):264–276, June 1987.

[14] G. F. A. LHospital. *Analyse des infiniment petits, Pour lintelligence des lignes courbes*. LImprimerie Royale, Paris, 1696.

[15] Laura I. Meikle and Jacques D. Fleuriot. Formalizing Hilberts Grundlagen in Isabelle/Isar. In David Basin and Burkhart Wolff, editors, *Theorem Proving in Higher Order Logics*, Lecture Notes in Computer Science, pages 319–334. Springer Berlin Heidelberg, 2003.

[16] Lawrence C. Paulson. Isabelle: The next 700 theorem provers. *CoRR*, cs.LO/9301106, 1993.

[17] Lawrence C Paulsson and Jasmin C Blanchette. Three years of experience with Sledgehammer, a practical link between automatic and interactive theorem provers. In *Procof the 8th International Workshop on the Implementation of Logics (IWIL-2010), Yogyakarta, Indonesia*, volume 2, 2012.

[18] Abraham Robinson and W A J Luxemburg. *Non-standard analysis*. Princeton landmarks in mathematics and physics. Princeton University Press, Princeton, NJ, 1996.

[19] Michael Spivak. *Calculus*. Publish or Perish, Inc., Houston, Tex., 4 edition edition, January 2008.

[20] Markus M Wenzel. *Isabelle/Isar—a versatile environment for human-readable formal proof documents*. PhD thesis, Technische Universität München, 2002.

[21] Freek Wiedijk. The de Bruijn factor. `http://www.cs.ru.nl/F.Wiedijk/factor/factor.pdf`, 2000.

Appendix A

180 - Differential of $\ln\, x$

A.1 Internal Function

lemma *ln-powser-internalfun*: $\bigwedge$ *x dx* :: *hypreal*. *dx* $\in$ *Infinitesimal* $\Longrightarrow$
  $(\lambda n.\ (-1)$ *pow* $n * (1/$*hypreal-of-hypnat*$(n + 1)) *$
        (dx/x) *pow* $(n + 1)) \in$ *InternalFuns*
proof $-$
 fix *x dx* :: *hypreal*
 assume *dx* $\in$ *Infinitesimal*
 have $(\lambda n.\ (-1$::*hypreal*) *pow* $n) \in$ *InternalFuns*
  using *InternalFuns-pow*[*of* $(\lambda n.\ -1)$ $(\lambda n.\ n)$] **by** *auto*
 moreover have $(\lambda n.\ 1/$*hypreal-of-hypnat*$(n + 1)) \in$ *InternalFuns*
 proof $-$
  have $(\lambda n.\ n + 1) \in$ *InternalFuns*
   using *InternalFuns-add*[*of* $(\lambda n.\ n)$ $(\lambda n.\ 1)$] **by** *auto*
  hence $(\lambda n.$ *hypreal-of-hypnat*$(n + 1)) \in$ *InternalFuns*
   using *InternalFuns-of-hypnat*[*of* $(\lambda n.\ n + 1)$] **by** *auto*
  thus $(\lambda n.\ 1/$*hypreal-of-hypnat*$(n + 1)) \in$ *InternalFuns*
   using *InternalFuns-divide*[*of* $(\lambda n.\ 1)$
     $(\lambda n.$ *hypreal-of-hypnat*$(n + 1))$] **by** *auto*
 qed
 moreover have $(\lambda n.\ (dx/x)$ *pow* $(n + 1)) \in$ *InternalFuns*
  using *InternalFuns-pow*[*of* $(\lambda n.\ dx/x)$ $(\lambda n.\ n + 1)$]
     InternalFuns-add[*of* $(\lambda n.\ n)$ $(\lambda n.\ 1)$] **by** *auto*
 ultimately show $(\lambda n.\ (-1)$*pow* $n * (1/$*hypreal-of-hypnat*$(n + 1)) *$
          (dx/x)*pow*$(n{+}1)) \in$ *InternalFuns*
  using *InternalFuns-mult* **by** *blast*
qed

A.2 Remainder Sum is Infinitesimal

lemma *remainder-infinitesimal*:
assumes *dx* ∈ *Infinitesimal* **and** *x* > *0*
and (∗*f*∗ *ln*) ((*star-of x*) + *dx*) − (∗*f*∗ *ln*) (*star-of x*)
≈ *hypersetsum* (λ*n*. ((−*1*) *pow n*)∗(*1*/*hypreal-of-hypnat*(*n* + *1*))∗
((*dx*/(*star-of x*)) *pow* (*n* + *1*))) {*1*..*N*} + (*dx* / *star-of x*)
shows *hypersetsum* (λ*n*. ((−*1*) *pow n*)∗(*1*/*hypreal-of-hypnat*(*n* + *1*))
∗((*dx*/*star-of x*) *pow* (*n* + *1*))) {*1*..*N*} ∈ *Infinitesimal*
proof −
have *isNSCont ln x* **using** *assms*(*2*)
using *isCont-ln*[*of x*] *isCont-isNSCont* **by** *auto*
hence (∗*f*∗ *ln*) ((*star-of x*) + *dx*) ≈ (∗*f*∗ *ln*) (*star-of x*)
by (*simp add*: *assms*(*1*) *isNSContD Infinitesimal-add-approx-self approx-sym*)
hence (∗*f*∗ *ln*) ((*star-of x*) + *dx*) − (∗*f*∗ *ln*) (*star-of x*) ∈ *Infinitesimal*
using *bex-Infinitesimal-iff* **by** *auto*
hence *hypersetsum* (λ*n*. (−*1*) *pow n* ∗ (*1*/*hypreal-of-hypnat*(*n* + *1*)) ∗
((*dx*/(*star-of x*)) *pow* (*n* + *1*))) {*1*..*N*} + (*dx* / *star-of x*) ∈ *Infinitesimal*
using *assms* **by** (*smt approx-sym approx-trans mem-infmal-iff*)
moreover have (*dx* / *star-of x*) ∈ *Infinitesimal*
using *assms*(*1*,*2*) *infinitesimal-divide* **by** *auto*
ultimately show
hypersetsum (λ*n*. (−*1*) *pow n* ∗ (*1*/*hypreal-of-hypnat*(*n* + *1*)) ∗
((*dx*/(*star-of x*)) *pow* (*n* + *1*))) {*1*..*N*} ∈ *Infinitesimal*
using *Infinitesimal-add-approx-self approx-trans mem-infmal-iff* **by** *blast*
qed

Appendix B

181 - Differential of $\ln(p\ x)$

```
lemma dy-ln-px:
  fixes dx :: hypreal and x::real and p::real ⇒ real
  assumes dx ∈ Infinitesimal − {0}
  and isNSCont p x   and x > 0
  and ( ∗f∗ p) (star-of x) > 0
shows ( ∗f∗ ln ) (( ∗f∗ p) ((star-of x) + dx))
                    − ( ∗f∗ ln ) (( ∗f∗ p) (star-of x))
    ≈ (( ∗f∗ p) ((star-of x) + dx)  − ( ∗f∗ p) (star-of x))
                          / (( ∗f∗ p) (star-of x))
proof −
  obtain dp where DP:dp = ( ∗f∗ p) ((star-of x) + dx) − ( ∗f∗ p) (star-of x)
    by auto
  have DPI: dp ∈ Infinitesimal
    using DP assms isNSContD Infinitesimal-add-approx-self
      approx-sym bex-Infinitesimal-iff by fastforce
  obtain px pdx where PX: px =  p x
    and PDX: pdx = ( ∗f∗ p) ((star-of x) + dx)
    by auto
  have ( ∗f∗ ln ) (( ∗f∗ p) ((star-of x) + dx))
                    − ( ∗f∗ ln ) (( ∗f∗ p) (star-of x))
    = ( ∗f∗ ln) (star-of px + dp) − ( ∗f∗ ln) (star-of px)
    using DP PX by auto
  moreover have ( ∗f∗ ln) (star-of px + dp) − ( ∗f∗ ln)(star-of px)
                          ≈ dp /(star-of px)
  proof −
    have A: px > 0
      using PX assms by auto
    thus ?thesis
    proof (cases dp = 0)
      case True
      then have (star-of px + dp) = (star-of px)
        using DP by auto
      hence ( ∗f∗ ln) (star-of px + dp) = ( ∗f∗ ln)(star-of px)
        using DP by auto
```

then show *?thesis* **using** *True* **by** *auto*
next
case *False*
hence $dp \in Infinitesimal - \{0\}$ **using** *DPI DP False* **by** *auto*
thus $(\ *f*\ ln)\ (star\text{-}of\ px + dp) - (\ *f*\ ln)\ (star\text{-}of\ px)$
$\approx dp\ /\ (star\text{-}of\ px)$
using *dy-ln-x[of dp px] assms A* **by** *auto*
qed
qed
ultimately show $(\ *f*\ ln\ )\ ((\ *f*\ p)\ ((star\text{-}of\ x) + dx))$
$- (\ *f*\ ln\ )\ ((\ *f*\ p)\ (star\text{-}of\ x))$
$\approx ((\ *f*\ p)\ ((star\text{-}of\ x) + dx)\ - (\ *f*\ p)\ (star\text{-}of\ x))$
$/\ ((\ *f*\ p)\ (star\text{-}of\ x))$
using *DP PX* **by** *auto*
qed

Appendix C

183 - Products of Fuctions of x

C.1 Including Division

lemma *dy-ln-mult-div*:
  fixes *p q r s* :: *real* **and** *dp dq dr ds* :: *hypreal*
  assumes *dp* ∈ *Infinitesimal* − {*0*} **and** *dq* ∈ *Infinitesimal* − {*0*}
    and *dr* ∈ *Infinitesimal* − {*0*} **and** *ds* ∈ *Infinitesimal* − {*0*}
    and *p* > *0* **and** *q* > *0* **and** *r* > *0* **and** *s* > *0*
  shows (∗*f*∗ *ln*)(((*star-of p*+*dp*)∗(*star-of q*+*dq*))
      / ((*star-of r*+*dr*)∗(*star-of s*+*ds*)))
    − (∗*f*∗ *ln*) ((*star-of p*∗*star-of q*) / (*star-of r*∗*star-of s*))
    ≈ (*dp*/*star-of p*) + (*dq*/*star-of q*) − (*dr*/*star-of r*) − (*ds*/*star-of s*)
proof −
 have *P*: *star-of p*+*dp* > *0*
  using *assms*(*1*,*5*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
 have *Q*: *star-of q*+*dq*>*0*
  using *assms*(*2*,*6*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
 have *R*: *star-of r*+*dr*>*0*
  using *assms*(*3*,*7*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
 have *S*: *star-of s*+*ds* > *0*
  using *assms*(*4*,*8*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*

 obtain *dy* **where**
  DY: *dy* = (∗*f*∗ *ln*)(((*star-of p*+*dp*)∗(*star-of q*+*dq*))
      / ((*star-of r*+*dr*)∗(*star-of s*+*ds*)))
    − (∗*f*∗ *ln*) ((*star-of p*∗*star-of q*) / (*star-of r*∗*star-of s*))
  by *auto*
 have (∗*f*∗ *ln*) ((*star-of p*∗*star-of q*) / (*star-of r*∗*star-of s*))
   = (∗*f*∗ *ln*) (*star-of p*) + (∗*f*∗ *ln*) (*star-of q*)
     − (∗*f*∗ *ln*) (*star-of r*) − (∗*f*∗ *ln*) (*star-of s*)
  by (*simp add*: *assms*(*5*−*8*) *starfun-ln-div starfun-ln-mult*)
 moreover have (∗*f*∗ *ln*)(((*star-of p*+*dp*)∗(*star-of q*+*dq*))
     / ((*star-of r*+*dr*)∗(*star-of s*+*ds*)))
   = (∗*f*∗ *ln*) (*star-of p*+*dp*) + (∗*f*∗ *ln*) (*star-of q*+*dq*)

$- (\ast f\ast\ ln)\ (\text{star-of}\ r+dr) - (\ast f\ast\ ln)\ (\text{star-of}\ s+ds)$
using *P Q R S starfun-ln-div starfun-ln-mult* **by** *auto*
ultimately have $DY'{:}dy = (((\ast f\ast\ ln)\ (\text{star-of}\ p+dp) - (\ast f\ast\ ln)\ (\text{star-of}\ p))$
$+ ((\ast f\ast\ ln)\ (\text{star-of}\ q+dq) - (\ast f\ast\ ln)\ (\text{star-of}\ q)))$
$- ((\ast f\ast\ ln)\ (\text{star-of}\ r+dr) - (\ast f\ast\ ln)\ (\text{star-of}\ r))$
$- ((\ast f\ast\ ln)\ (\text{star-of}\ s+ds) - (\ast f\ast\ ln)\ (\text{star-of}\ s))$
using *DY* **by** *auto*
have P: $(\ast f\ast\ ln)(\text{star-of}\ p+dp) - (\ast f\ast\ ln)\ (\text{star-of}\ p) \approx dp/\text{star-of}\ p$
and Q: $(\ast f\ast\ ln)(\text{star-of}\ q+dq) - (\ast f\ast\ ln)\ (\text{star-of}\ q) \approx dq/\text{star-of}\ q$
and R: $(\ast f\ast\ ln)(\text{star-of}\ r+dr) - (\ast f\ast\ ln)\ (\text{star-of}\ r) \approx dr/\text{star-of}\ r$
and S: $(\ast f\ast\ ln)(\text{star-of}\ s+ds) - (\ast f\ast\ ln)\ (\text{star-of}\ s) \approx ds/\text{star-of}\ s$
using *dy-ln-x assms* **by** *auto*
have $((\ast f\ast\ ln)\ (\text{star-of}\ p+dp) - (\ast f\ast\ ln)\ (\text{star-of}\ p))$
$+ ((\ast f\ast\ ln)\ (\text{star-of}\ q+dq) - (\ast f\ast\ ln)\ (\text{star-of}\ q))$
$\approx dp/\text{star-of}\ p + dq/\text{star-of}\ q$
using *approx-add P Q* **by** *auto*
hence $((\ast f\ast\ ln)\ (\text{star-of}\ p+dp) - (\ast f\ast\ ln)\ (\text{star-of}\ p))$
$+ ((\ast f\ast\ ln)\ (\text{star-of}\ q+dq) - (\ast f\ast\ ln)\ (\text{star-of}\ q))$
$- ((\ast f\ast\ ln)\ (\text{star-of}\ r+dr) - (\ast f\ast\ ln)\ (\text{star-of}\ r))$
$\approx dp/\text{star-of}\ p + dq/\text{star-of}\ q - (dr/\text{star-of}\ r)$
using *approx-add approx-minus-cancel approx-diff R* **by** *auto*
hence $((\ast f\ast\ ln)\ (\text{star-of}\ p+dp) - (\ast f\ast\ ln)\ (\text{star-of}\ p))$
$+ ((\ast f\ast\ ln)\ (\text{star-of}\ q+dq) - (\ast f\ast\ ln)\ (\text{star-of}\ q))$
$- ((\ast f\ast\ ln)\ (\text{star-of}\ r+dr) - (\ast f\ast\ ln)\ (\text{star-of}\ r))$
$- ((\ast f\ast\ ln)\ (\text{star-of}\ s+ds) - (\ast f\ast\ ln)\ (\text{star-of}\ s))$
$\approx dp/\text{star-of}\ p + dq/\text{star-of}\ q - (dr/\text{star-of}\ r) - (ds/\text{star-of}\ s)$
using *approx-add approx-minus-cancel approx-diff S* **by** *auto*
thus $(\ast f\ast\ ln)(((\text{star-of}\ p+dp)\ast(\text{star-of}\ q+dq))$
$/\ ((\text{star-of}\ r+dr)\ast(\text{star-of}\ s+ds)))$
$- (\ast f\ast\ ln)\ ((\text{star-of}\ p\ast\text{star-of}\ q)\ /\ (\text{star-of}\ r\ast\text{star-of}\ s))$
$\approx dp/\text{star-of}\ p + dq/\text{star-of}\ q - (dr/\text{star-of}\ r) - (ds/\text{star-of}\ s)$
using $DY'\ DY$ **by** *auto*
qed

C.2 Including Exponentiation

lemma *dy-ln-mult-div-pow*:
fixes *p q r s* :: *real* **and** *dp dq dr ds* :: *hypreal*
assumes $dp \in Infinitesimal - \{0\}$ **and** $dq \in Infinitesimal - \{0\}$
and $dr \in Infinitesimal - \{0\}$ **and** $ds \in Infinitesimal - \{0\}$
and $p > 0$ **and** $q > 0$ **and** $r > 0$ **and** $s > 0$
and *hypreal-of-hypnat*(m) $\in$ *HFinite* **and** *hypreal-of-hypnat*(n) $\in$ *HFinite*
and *hypreal-of-hypnat*(μ) $\in$ *HFinite* **and** *hypreal-of-hypnat*(ν) $\in$ *HFinite*
shows (**f* ln*)((((*star-of p*+*dp*) *pow m*)*((*star-of q*+*dq*) *pow n*))
/ (((*star-of r*+*dr*) *pow* μ)*((*star-of s*+*ds*) *pow* ν)))
− (**f* ln*) (((*star-of p pow m*)*(*star-of q pow n*))
/ ((*star-of r pow* μ)*(*star-of s pow* ν)))
≈ (*hypreal-of-hypnat*(m) * *dp*/*star-of p*)
+ (*hypreal-of-hypnat*(n) * *dq*/*star-of q*)
− (*hypreal-of-hypnat*(μ) * *dr*/*star-of r*)
− (*hypreal-of-hypnat*(ν) * *ds*/*star-of s*)
proof −
have *P*: *star-of p*+*dp* > *0*
using *assms*(*1*,*5*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
have *Q*: *star-of q*+*dq*>*0*
using *assms*(*2*,*6*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
have *R*: *star-of r*+*dr*>*0*
using *assms*(*3*,*7*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
have *S*: *star-of s*+*ds* > *0*
using *assms*(*4*,*8*)*Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
obtain *dy* **where**
DY: *dy* = (**f* ln*)((((*star-of p*+*dp*) *pow m*)*((*star-of q*+*dq*) *pow n*))
/ (((*star-of r*+*dr*) *pow* μ)*((*star-of s*+*ds*) *pow* ν)))
− (**f* ln*) (((*star-of p pow m*)*(*star-of q pow n*))
/ ((*star-of r pow* μ)*(*star-of s pow* ν)))
by *auto*
have (**f* ln*) (((*star-of p pow m*)*(*star-of q pow n*))
/ ((*star-of r pow* μ)*(*star-of s pow* ν)))
= (*hypreal-of-hypnat*(m) * (**f* ln*) (*star-of p*))
+ (*hypreal-of-hypnat*(n) * (**f* ln*) (*star-of q*))
− (*hypreal-of-hypnat*(μ) * (**f* ln*) (*star-of r*))
− (*hypreal-of-hypnat*(ν) * (**f* ln*) (*star-of s*))
using *assms starfun-ln-div starfun-ln-mult starfun-ln-pow*
by (*simp add*: *hyperpow-gt-zero*)
moreover have (**f* ln*)((((*star-of p*+*dp*) *pow m*)*((*star-of q*+*dq*) *pow n*))
/ (((*star-of r*+*dr*) *pow* μ)*((*star-of s*+*ds*) *pow* ν)))
= (*hypreal-of-hypnat*(m) * (**f* ln*) (*star-of p*+*dp*))
+ (*hypreal-of-hypnat*(n) * (**f* ln*) (*star-of q*+*dq*))
− (*hypreal-of-hypnat*(μ) * (**f* ln*) (*star-of r*+*dr*))
− (*hypreal-of-hypnat*(ν) * (**f* ln*) (*star-of s*+*ds*))
using *P Q R S starfun-ln-div starfun-ln-mult starfun-ln-pow*
by (*simp add*: *hyperpow-gt-zero*)
ultimately have *dy* =

$((hypreal\text{-}of\text{-}hypnat(m) * (*f* ln)\ (star\text{-}of\ p{+}dp))$
$- (hypreal\text{-}of\text{-}hypnat(m) * (*f* ln)\ (star\text{-}of\ p)))$
$+ ((hypreal\text{-}of\text{-}hypnat(n) * (*f* ln)\ (star\text{-}of\ q{+}dq))$
$- (hypreal\text{-}of\text{-}hypnat(n) * (*f* ln)\ (star\text{-}of\ q)))$
$- ((hypreal\text{-}of\text{-}hypnat(\mu) * (*f* ln)\ (star\text{-}of\ r{+}dr))$
$- (hypreal\text{-}of\text{-}hypnat(\mu) * (*f* ln)\ (star\text{-}of\ r)))$
$- ((hypreal\text{-}of\text{-}hypnat(\nu) * (*f* ln)\ (star\text{-}of\ s{+}ds))$
$- (hypreal\text{-}of\text{-}hypnat(\nu) * (*f* ln)\ (star\text{-}of\ s)))$

using *DY* **by** *auto*

moreover have $\bigwedge x\ dx\ n.((hypreal\text{-}of\text{-}hypnat(n)*(*f* ln)(star\text{-}of\ x{+}dx))$
$- (hypreal\text{-}of\text{-}hypnat(n)*(*f* ln)\ (star\text{-}of\ x)))$
$= (hypreal\text{-}of\text{-}hypnat(n) *$
$((*f* ln)\ (star\text{-}of\ x{+}dx) - (*f* ln)\ (star\text{-}of\ x)))$

by (*simp add*: *right-diff-distrib′*)

ultimately have

$A{:}dy = (hypreal\text{-}of\text{-}hypnat(m) *$
$((*f* ln)\ (star\text{-}of\ p{+}dp) - (*f* ln)\ (star\text{-}of\ p)))$
$+ (hypreal\text{-}of\text{-}hypnat(n) *$
$((*f* ln)\ (star\text{-}of\ q{+}dq) - (*f* ln)\ (star\text{-}of\ q)))$
$- (hypreal\text{-}of\text{-}hypnat(\mu) *$
$((*f* ln)\ (star\text{-}of\ r{+}dr) - (*f* ln)\ (star\text{-}of\ r)))$
$- (hypreal\text{-}of\text{-}hypnat(\nu) *$
$((*f* ln)\ (star\text{-}of\ s{+}ds) - (*f* ln)\ (star\text{-}of\ s)))$

by *auto*

have *P*: $(*f* ln)(star\text{-}of\ p{+}dp){-}(*f* ln)\ (star\text{-}of\ p) \approx dp/star\text{-}of\ p$
and *Q*: $(*f* ln)(star\text{-}of\ q{+}dq){-}(*f* ln)\ (star\text{-}of\ q) \approx dq/star\text{-}of\ q$
and *R*: $(*f* ln)(star\text{-}of\ r{+}dr){-}(*f* ln)\ (star\text{-}of\ r) \approx dr/star\text{-}of\ r$
and *S*: $(*f* ln)(star\text{-}of\ s{+}ds){-}(*f* ln)\ (star\text{-}of\ s) \approx ds/star\text{-}of\ s$

using *dy-ln-x assms* **by** *auto*

have *H*: $\bigwedge x\ dx\ n.(*f* ln)(star\text{-}of\ x{+}dx) - (*f* ln)\ (star\text{-}of\ x)$
$\approx dx/star\text{-}of\ x \implies hypreal\text{-}of\text{-}hypnat(n) \in HFinite$
$\implies (hypreal\text{-}of\text{-}hypnat(n) * ((*f* ln)\ (star\text{-}of\ x{+}dx)$
$- (*f* ln)\ (star\text{-}of\ x))) \approx (hypreal\text{-}of\text{-}hypnat(n) * (dx/star\text{-}of\ x))$

using *approx-mult2* **by** *blast*

have $(hypreal\text{-}of\text{-}hypnat(m) *$
$((*f* ln)\ (star\text{-}of\ p{+}dp) - (*f* ln)\ (star\text{-}of\ p)))$
$+ (hypreal\text{-}of\text{-}hypnat(n) *$
$((*f* ln)\ (star\text{-}of\ q{+}dq) - (*f* ln)\ (star\text{-}of\ q)))$
$\approx (hypreal\text{-}of\text{-}hypnat(m) * (dp/star\text{-}of\ p))$
$+ (hypreal\text{-}of\text{-}hypnat(n) * (dq/star\text{-}of\ q))$

using *approx-add P Q H assms* **by** *blast*

hence $(hypreal\text{-}of\text{-}hypnat(m) *$
$((*f* ln)\ (star\text{-}of\ p{+}dp) - (*f* ln)\ (star\text{-}of\ p)))$
$+ (hypreal\text{-}of\text{-}hypnat(n) *$
$((*f* ln)\ (star\text{-}of\ q{+}dq) - (*f* ln)\ (star\text{-}of\ q)))$
$- (hypreal\text{-}of\text{-}hypnat(\mu) *$
$((*f* ln)\ (star\text{-}of\ r{+}dr) - (*f* ln)\ (star\text{-}of\ r)))$
$\approx (hypreal\text{-}of\text{-}hypnat(m) * (dp/star\text{-}of\ p))$
$+ (hypreal\text{-}of\text{-}hypnat(n) * (dq/star\text{-}of\ q))$

$-$ (*hypreal-of-hypnat*(μ) ∗ (*dr*/*star-of r*))
using *approx-add*[*of* - (*hypreal-of-hypnat*(*m*) ∗ (*dp*/*star-of p*)) +
(*hypreal-of-hypnat*(*n*) ∗ (*dq*/*star-of q*)) (*hypreal-of-hypnat*(μ) ∗
((∗*f*∗ *ln*) (*star-of r*+*dr*) − (∗*f*∗ *ln*) (*star-of r*)))]
approx-minus-cancel approx-diff R H assms **by** *blast*
hence (*hypreal-of-hypnat*(*m*) ∗
((∗*f*∗ *ln*) (*star-of p*+*dp*) − (∗*f*∗ *ln*) (*star-of p*)))
\+ (*hypreal-of-hypnat*(*n*) ∗
((∗*f*∗ *ln*) (*star-of q*+*dq*) − (∗*f*∗ *ln*) (*star-of q*)))
− (*hypreal-of-hypnat*(μ) ∗
((∗*f*∗ *ln*) (*star-of r*+*dr*) − (∗*f*∗ *ln*) (*star-of r*)))
− (*hypreal-of-hypnat*(ν) ∗
((∗*f*∗ *ln*) (*star-of s*+*ds*) − (∗*f*∗ *ln*) (*star-of s*)))
≈ (*hypreal-of-hypnat*(*m*) ∗ (*dp*/*star-of p*))
\+ (*hypreal-of-hypnat*(*n*) ∗ (*dq*/*star-of q*))
− (*hypreal-of-hypnat*(μ) ∗ (*dr*/*star-of r*))
− (*hypreal-of-hypnat*(ν) ∗ (*ds*/*star-of s*))
using *approx-add*[*of* - (*hypreal-of-hypnat*(*m*) ∗ *dp*/*star-of p*) +
(*hypreal-of-hypnat*(*n*) ∗ *dq*/*star-of q*) − (*hypreal-of-hypnat*(μ) ∗
dr/*star-of r*) *hypreal-of-hypnat*(ν) ∗ ((∗*f*∗ *ln*) (*star-of s*+*ds*) −
(∗*f*∗ *ln*) (*star-of s*))] *approx-diff S H assms* **by** *blast*
thus *?thesis* **using** *DY A* **by** *auto*
qed

Appendix D

184 - Higher Differentials and Derivatives

D.1 dy is Nonzero

lemma *dy-neq-zero*:
 fixes x :: *real* **and** dx :: *hypreal*
 assumes $dx \in Infinitesimal - \{0\}$ **and** $x > 0$
 shows $dy\ ln\ x\ dx \neq 0$
proof
 have *XDX*: $(star\text{-}of\ x) + dx > 0$ **using** *assms*
  using *Infinitesimal-add-hypreal-of-real-less* **by** *fastforce*
 have *DY*: $dy\ ln\ x\ dx = (\ *f*\ ln\ )\ ((star\text{-}of\ x) + dx) - (\ *f*\ ln\ )\ (star\text{-}of\ x)$
  by (*simp add*: *seq-apply-function-def seq-delta-def*)
 assume $dy\ ln\ x\ dx = 0$
 hence $(\ *f*\ ln\ )\ ((star\text{-}of\ x) + dx) = (\ *f*\ ln\ )\ (star\text{-}of\ x)$
  using *assms DY* **by** *auto*
 moreover have $(\ *f*\ exp)\ ((\ *f*\ ln)\ ((star\text{-}of\ x) + dx)) = (star\text{-}of\ x) + dx$
  using *starfun-exp-ln-iff*[*of* $(star\text{-}of\ x) + dx$] *assms XDX* **by** *auto*
 moreover have $(\ *f*\ exp)\ ((\ *f*\ ln)\ (star\text{-}of\ x)) = (star\text{-}of\ x)$
  using *assms*(2) *starfun-exp-ln-iff* **by** *auto*
 ultimately have $(star\text{-}of\ x) = (star\text{-}of\ x) + dx$ **by** *auto*
 thus *False*
  using *assms* **by** *auto*
qed

D.2 dy is Infinitesimal

lemma *dy-infinitesimal*:
fixes *x* :: *real* **and** *dx* :: *hypreal*
assumes $dx \in Infinitesimal - \{0\}$ **and** $x > 0$
shows $dy\ ln\ x\ dx \in Infinitesimal$
proof −
have $dx/(star\text{-}of\ x) \in Infinitesimal$ **using** *assms*
using *infinitesimal-divide* **by** *auto*
moreover have $dy\ ln\ x\ dx =$
$(\ {*}f{*}\ ln\ )\ ((star\text{-}of\ x) + dx) - (\ {*}f{*}\ ln\ )\ (star\text{-}of\ x)$
by (*simp add*: *seq-apply-function-def seq-delta-def*)
moreover have $(\ {*}f{*}\ ln\ )\ ((star\text{-}of\ x) + dx) - (\ {*}f{*}\ ln\ )\ (star\text{-}of\ x)$
$\approx\ dx\ /\ (star\text{-}of\ x)$
using *assms dy-ln-x* **by** *auto*
ultimately show *?thesis*
using *approx-trans mem-infmal-iff* **by** *auto*
qed

D.3 Factor from Infinitely Close Statement

lemma *approx-factor*:
fixes A B :: *hypreal*
assumes $A \approx B$ **and** $A \in \mathit{HFinite}$ **and** $B \in \mathit{HFinite}$
and $A \notin \mathit{Infinitesimal} \vee B \notin \mathit{Infinitesimal}$
shows $\exists\ a.\ a \approx 1 \wedge a * A = B$

proof $-$
from *assms* $(1, 4)$ **have** *A1*: $A \notin \mathit{Infinitesimal}$ **and** *B1*: $B \notin \mathit{Infinitesimal}$
using *approx-trans3 mem-infmal-iff approx-sym* **by** *blast*$+$
hence *C*: $st(A) = st(B)$ **and** *NZ*: $st(A) \neq 0$
using *assms st-eq-approx-iff HFinite-st-Infinitesimal-add*
by $(auto, fastforce)$
then obtain a **and** b **where** *A2*: $a \in \mathit{Infinitesimal}$ **and** *B2*: $b \in \mathit{Infinitesimal}$
and $A = st(A) + a$ **and** $B = st(B) + b$
using *HFinite-st-Infinitesimal-add assms* **by** *meson*
hence $st(A) = A - a$ **and** $st(B) = B - b$ **by** *auto*
hence *A3*: $st(A) = A * (1 - a/A)$ **and** *B3*:$st(B) = B * (1 - b/B)$
by (*simp add*: *A1 B1 not-Infinitesimal-not-zero right-diff-distrib*$'$)$+$
hence $A * (1 - a/A) = B * (1 - b/B)$ **using** *C* **by** *auto*
hence $A * (1 - a/A)\ /\ (1 - b/B) = B$ **using** *A3 NZ* **by** *auto*
moreover have $(1 - a/A)\ /\ (1 - b/B) \approx 1$
proof $-$
have $a/A \in \mathit{Infinitesimal}$ **and** $b/B \in \mathit{Infinitesimal}$
using *assms A1 A2 B1 B2*
by (*simp add*: *HFinite-inverse Infinitesimal-HFinite-mult divide-inverse*)$+$
hence $(1 - a/A) \approx 1$ **and** $(1 - b/B) \approx 1$
using *Infinitesimal-approx-minus* **by** *fastforce*$+$
moreover have *A4*: $(1 - a/A) \in \mathit{HFinite}$ **and** *B4*:$(1 - b/B) \in \mathit{HFinite}$
using *HFinite-1 approx-HFinite*[*of 1*] *approx-sym calculation* **by** *blast*$+$
ultimately have *A5*: $st(1 - a/A) = 1$ **and** *B5*:$st(1 - b/B) = 1$
using *st-eq-approx-iff st-1* **by** *auto*
hence $st(1 - a/A)\ /\ st(1 - b/B) = 1$ **by** *auto*
hence $st((1 - a/A)\ /\ (1 - b/B)) = 1$
using *A4 B4 st-divide* **by** *auto*
moreover have $(1 - a/A)\ /\ (1 - b/B) \in \mathit{HFinite}$ **using** *A4 B4 B5*
HFinite-mult[*of* $(1 - a/A)$ *inverse* $(1 - b/B)$] *divide-inverse*
one-neq-zero HFinite-inverse[*of* $(1 - b/B)$] *HFinite-mult*
by (*metis st-Infinitesimal*)
ultimately show *?thesis* **using** *st-eq-approx st-1* **by** *auto*
qed
ultimately have $((1 - a/A)\ /\ (1 - b/B)) \approx 1$
$\wedge\ ((1 - a/A)\ /\ (1 - b/B)) * A = B$
by (*simp add*: *semiring-normalization-rules*(7))
thus *?thesis* **by** *auto*
qed