Now f is continuous (exercise!)

R.D. Arthan
Lemma 1 Ltd.
2nd Floor, 31A Chain Street
Reading UK RG1 2HX
rda@lemma-one.com

February 22, 2012

Abstract

A recurring proof obligation in modern mathematics is to show that a given function is a morphism in some category. For example, consider the function $f(x) = e^{2\pi ix}$. If we are interested in algebraic properties of f, we will want to know that it is a homomorphism from the additive group of real numbers to the multiplicative group of non-zero complex numbers, while if we are interested in topological properties, we will want to know that it is continuous with respect to the appropriate topologies. As the title reminds us, in traditional mathematical practice, discharging such proof obligations is often routine and they are left as exercises for the reader. When we are using a mechanized theorem-prover to formalise mathematics, these exercises are often tedious to carry out manually and we can very reasonably expect automatic support to help us find the proofs. Some work on this kind of automation for specific structures has already been done. The formalisations of analysis in HOL Light, ProofPower and PVS, for example, include tools that help to prove the continuity of algebraic combinations of continuous functions on the real line. In this talk, I will report on some recent work on generalisations of this kind of tool.

I will set the scene with a brief discussion of formalisations of categories such as **Grp**, **Top**, **Rng** and $\mathbb{R}-$**Vec** that occur so frequently in mathematical practice. The goal is not to formalise any category theory, but rather to use some basic category-theoretic concepts to help structure our thinking about the formalisation. The categories in question generally have a rather different flavour from the categories that commonly occur in computer science. For example, apart from the category of sets, cartesian closed categories are rather rare in mathematics. The aim of the work was to provide automation that helps one prove that a given combination of functions is a morphism in one of the categories of interest. I will describe some prototype algorithms and tools that give a framework for finding these proofs. This will be illustrated with examples drawn from the ProofPower formalisations of topological and metric spaces.