Towards Specifying Symbolic Computation*

Jacques Carette and William M. Farmer

Computing and Software, McMaster University, Canada
http://www.cas.mcmaster.ca/~carette
http://imps.mcmaster.ca/wmfarmer

Abstract. ??

1 Introduction

2 Simplifying Expressions denoting Rational Functions

2.1 The Problem

Let $f = \lambda\, x : \mathbb{Q}\, .\, R(x)$ be an expression that denotes a function of type $\mathbb{Q} \to \mathbb{Q}$. Furthermore, let $R(x)$, as a syntactic expression, denote a member of the field $\mathbb{Q}(x)$ consisting of *rational expressions* of the form $\frac{P(x)}{Q(x)}$ where $P(x)$ and $Q(x)$ are polynomials in x. A function that can be represented by an expression like f whose body is a rational expression is called a *rational function*.

$R(x)$ could be a very complicated expression build from the primitive components of the field $\mathbb{Q}(x)$: $0, 1, +, *, -, {}^{-1}$. An obvious way to simplify f would be to simplify $R(x)$ syntactically as a rational expression in a meaning-preserving way. This operation, which we will call simpRatFun, is an example of a *syntax-based mathematical algorithm (SBMA)*. As an SBMA, simpRatFun works by manipulating syntactic expressions in a mathematically meaningful way.

SBMAs can be difficult to specify since they involve an interplay of syntax and semantics. We are interested in the following three questions:

1. What should be the specification of simpRatFun?
2. Can this specification be expressed in traditional logic?
3. How would this specification be expressed in a logic with undefinedness, quotation, and evaluation?

2.2 A Naive Specification

Let $R(x)$ be an expression that denotes a member of $\mathbb{Q}(x)$. The *normal form* of $R(x)$ is an expression $R'(x)$ of the form $\frac{P(x)}{Q(x)}$ such that $P(x)$ and $Q(x)$ are

* This research is supported by NSERC.

polynomials in standard form, $\frac{P(x)}{Q(x)}$ is in lowest terms, and $R(x)$ and $R'(x)$ both denote the same member of $\mathbb{Q}(x)$. For example, the normal form of

$$\frac{x^2+2x-1}{x^2-1} - \frac{x}{x-1}$$

is

$$\frac{1}{x+1}.$$

If we do not think too hard, we might be tempted to specify $\mathsf{simpRatFun}$ as follows:

> For all expressions $f = \lambda\, x : \mathbb{Q} \,.\, R(x)$ of type $\mathbb{Q} \rightarrow \mathbb{Q}$ where the expression $R(x)$ denotes a member of $\mathbb{Q}(x)$, $\mathsf{simpRatFun}(f) = \lambda\, x : \mathbb{Q} \,.\, R'(x)$ where $R'(x)$ is the normal form of $R(x)$.

Hence $\mathsf{simpRatFun}$ applied to

$$\lambda\, x : \mathbb{Q} \,.\, \frac{x^2+2x-1}{x^2-1} - \frac{x}{x-1}$$

should return

$$\lambda\, x : \mathbb{Q} \,.\, \frac{1}{x+1}.$$

This specification is essentially the same as Maple's normal operation that reduces the expression $\frac{x^2-2x-1}{x^2-1} - \frac{x}{x-1}$ to $\frac{1}{x+1}$.

Is this specification of $\mathsf{simpRatFun}$ correct? If so, f and $\mathsf{simpRatFun}(f)$ should denote the same function of type $\mathbb{Q} \rightarrow \mathbb{Q}$ for all expressions f that denote rational functions. Unfortunately, this is not the case. Let

$$f = \lambda\, x : \mathbb{Q} \,.\, \frac{x^2+2x-1}{x^2-1} - \frac{x}{x-1}.$$

Then $f(1)$ is obviously undefined, but $\mathsf{simpRatFun}(f)(1) = 1/2$. Hence $\mathsf{simpRatFun}$ is — as specified — not meaning preserving. What went wrong?

2.3 A Correct Specification

2.4 A Formalized Specification

3 Rational Expressions, Rational Functions

3.1 The Problem

Let e be an expression in the language $\mathcal{L}$ of the field $\mathbb{Q}(x)$, that is a well-formed expression built from the symbols $x, 0, 1, +, *, -, {}^{-1}$, elements of $\mathbb{Q}$ and parentheses (as necessary). We will take the liberty of using fractional notation for $^{-1}$ for greater readability. e can be something simple like $\frac{x^4-1}{x^2-1}$ or something more complicated like

$$\frac{\frac{1-x}{3/2x^{18}+x+17}}{frac19834 * x^{19393874} - 1/5} + 3 * x - \frac{12}{x}$$

We are taught that, like for members of $\mathbb{Q}$ (such as 5/15), there is a *normal form* for such expressions. This is typically defined to consist of an expression p/q for two polynomials p, q in $\mathbb{Q}[x]$. Typically, p and q are themselves put into normal form as well. The important property is that, as polynomials, we have that $gcd(p, q) = 1$. The same way that one would rarely tend to write 5/15, but rather the (equivalent) 1/3 instead. This is because one of the defining characteristics of the *field of fractions* of a ring R is that its elements can be written as quotients r/s of elements of R, and where $r_0/s_0 = r_1/s_1$ if and only if $r_0 * s_1 = r_1 * s_0$ in R.

All computer algebra systems implement functions that performs this task of *normalizing* elements of $\mathbb{Q}$ and $\mathbb{Q}(x)$. It is important to note that, in $\mathbb{Q}(x)$, $1 = \frac{x}{x}$, and our earlier example $\frac{x^4-1}{x^2-1} = x^2 + 1$. Let us name this algorithm normRatExpr. It should certainly have signature normRatExpr : $\mathcal{L} \to \mathcal{L}$ and satisfy the invariant normRatExpr$(e) = e$ as elements of $\mathbb{Q}(x)$ – in other words, it is *meaning preserving*.

However, the same expression e can also be interpreted as a *function* $f = \lambda x : \mathbb{Q} \, . \, e$. Such functions are typically called *rational functions*. But are the functions $\lambda x : \mathbb{Q} \, . \, x/x$ and $\lambda x : \mathbb{Q} \, . \, 1$ really equal? Your average CAS doesn't (quite) think so, unless you think that an error and 1 are equal! In a CAS, normalization functions operate on $\mathcal{L}$, and not on expressions like f. Of course, given some symbol y, $f(y)$ **is** in $\mathcal{L}$.

normRatExpr is an example of a *syntax-based mathematical algorithm (SBMA)*. SBMAs can be difficult to specify since they involve an interplay of syntax and semantics. We have to be careful of *which* semantics is used to guide the syntactic manipulations, as different semantics for the **same** expression can be inconsistent.

Note that merely giving the signature to normRatExprand saying that it is meaning preserving is a (tremendously) incomplete specification. We are thus interested in the following questions:

1. What should the specification of normRatExprbe?
2. What features of a logic are needed to express this adequately?
3. What features of a logic would make expressing this specification relatively straightforward?

3.2 A Naive Specification

Let e be an expression of $\mathcal{L}$ denoting a member of $\mathbb{Q}(x)$. The *normal form* of e is an expression e' of the form $\frac{p}{q}$ where p, q are polynomials in standard form,

$\mathsf{gcd}(p, q) = 1$ and $e = e' : \mathbb{Q}(x)$. For example, the normal form of

$$\frac{x^2 + 2x - 1}{x^2 - 1} - \frac{x}{x - 1}$$

is

$$\frac{1}{x + 1}.$$

...

3.3 A Correct Specification

3.4 A Formalized Specification

4 Conclusion

Todo list