Foundations of Computer Algebra: Project 1

Marc Moreno Maza
University of Western Ontario

CS 424b / CS556b – Winter 2005

January 7, 2008

0.1 Objectives

Determine the pratical efficiency of the Karatsuba's algorithm in AXIOM or MAPLE. That is, for a pair of ramdom univariate polynomials of degree n determine experimentally which algorithm to use between the Karatsuba's algorithm and the classical (quadratic) one. Obviously, this will depend not only on n but also on the coefficient ring. The following coefficients rings are recommended:

- the ring $\mathbb{Z}$ of the integer numbers,
- the ring $\mathbb{Z}/p\mathbb{Z}$ of the integer modulo p for $p \in \{11, 101, 1009, 10007, 100003\}$,
- the ring of univariate polynomials over $\mathbb{Z}$,
- the ring of univariate polynomials over $\mathbb{Z}/p\mathbb{Z}$ for $p \in \{11, 101, 1009, 10007, 100003\}$.

These choices will become clear later.

If you choose to work with MAPLE, you must implement your own univariate polynomial arithmetic by means of a canonical representation. To do do, you can adapt the file `polynomials.input` which implements multivariate polynomial arithmetic by means of a canonical representation. This is needed since in your implementation of the Karatsuba's algorithm you must control which algorithm is used during the recursive calls. If you would be using the built-in polynomial multiplication of MAPLE, then you would not know. As you can guess, the Maple polynomial multiplication already uses the Karatsuba's algorithm and other tricks.

0.2 Method

You must implement a generator of random univariate polynomials parametrized by the desired degree and the coefficient ring. In Maple

- there is already a generator of random univariate polynomials,
- and you may choose to write one generator per desired coefficient ring.

In AXIOM

- there is less support for generating random polynomials,
- but it is still quite easy.

For each coefficient ring, the degree n should vary from 1 to a value large enough (typically 200) in order to show that the Karatsuba's algorithm is asymptotically faster. For each value of n, it is recommended

- to pick several random pairs and run both algorithms for this pair,
- to compute the average time for each algorithm.

Both Maple and AXIOM have graphical facilities that you can use for displaying your experimental results. You may also use `gnuplot` or any other tool that you are familiar with.

0.3 Report

A postscript or a PDF document reporting on your experiments. Slides would be perfect. Note that implementing the Karatsuba's algorithm is quite easy. But making it really efficient for polynomials with low degree requires some care. One difficult point is to control the space complexity (because of the recursive calls). Each student will present her(his) results during a short (about 10 min) presentation in class.

0.4 Possible extensions

In order to do a more advanced and challenging work you could look at the following problems:

- For large polynomials, is it worth cutting each polynomial in 3 (or more) rather than 2 peices?
- Does the Karatsuba's trick apply for univariate polynomials which have matrix coefficients? If yes, try to do some experiments.
- ...

References

[GG99] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, 1999.