

Re: how to factor gaussian integers ?

Source: <http://sci.tech-archive.net/Archive/sci.math/2007-09/msg03874.html>

- *From:* Bill Dubuque <wgd@xxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* 17 Sep 2007 23:05:53 -0400
-

Phil Carmody <thefatphil_demunged@xxxxxxxxxxxxx> writes:

Bill Dubuque <wgd@xxxxxxxxxxxxxxxxxxxxxx> writes:

Pubkeybreaker <pubkeybreaker@xxxxxxx> wrote:

Bill Dubuque wrote:

Arturo Magidin
<magidin@xxxxxxxxxxxxxxxxxxxxxx> wrote:

There are very efficient algorithms for factoring in integral domains with only finitely many primes.

Most certainly not. If that were true then one could factor any integer N "efficiently" in the localization of \mathbb{Z} whose primes are precisely those integer primes smaller than N .

Let D be the set of all rational numbers that can be written as a/b , with a and b integers, and b relatively prime to, say, 2, 3, and 5.

This is an integral domain. The only primes in this domains are 2, 3, and 5.

Given any element x/y of D ,

Re: how to factor gaussian integers ?

with $\gcd(x,y)=1$, find the highest power of 2, of 3, and of 5 that divide x (which can be done both easily and efficiently). Write $x = 2^a \cdot 3^b \cdot 5^c \cdot e$, with $\gcd(e,30)=1$. Then x/y factors in D as $(e/y) \cdot 2^a \cdot 3^b \cdot 5^c$. Both easy and efficient.

Same idea if you have an integral domain with only finitely many primes.

That's simply trial division. Certainly easy, but hardly efficient.

Huh? The method is polynomial in the length of x . This is efficient in any measure of complexity.

Perhaps there is some confusion due to imprecision. Precisely what algorithm do you believe is efficient?

The same as everyone else. It's just that he's viewing efficiency to mean the asymptotic Big-Oh of the algorithm. As the trial division factor list never changes size, the algorithm complexity is barely worse than a GCD computation, i.e. poly in the input size (number to be factored).

This asymptotic behaviour in no way corresponds to whether the algorithm is efficient to implement in the real world.

A simple equivocation on 'efficient'.

That was precisely my point – to highlight the artificiality of this particular interpretation of "very efficient algorithms", lest a novice be misled to believe that localizations possess some nontrivial innate structure that simplifies factorization.

—Bill Dubuque