

Re: Factorisation algorithms

Source: <http://sci.tech-archive.net/Archive/sci.math/2007-10/msg02855.html>

- *From:* matt271829-news@xxxxxxxxxxx
 - *Date:* Tue, 16 Oct 2007 18:33:45 -0700
-

On Oct 17, 12:29 am, i...@xxxxxxxx (Randy Hudson) wrote:

In article <1192564600.554428.131...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>,

<matt271829-n...@xxxxxxxx> wrote:

Has anyone ever proved any theoretical bounds on the efficiency of a general integer factorisation algorithm? Is it still, as far as anyone knows, possible that a really spectacular advance might be made in this field?

Upper bounds:

Knuth cites John Douglas Dixon for an algorithm with a running time proven to be $O(N^{f(N)})$ with $f(N)$ approaching zero as N increases: specifically, $f(N)$ approaches $\sqrt{\ln(\ln(N)) * 8 / \ln(N)}$. Knuth gives the publication date as 1978, but I don't see a full cite to the paper.

He also cites Shamir, *Inf. Proc. Letters* 8 (1979), pp 28-31, for a demonstration that, if the processing time for an arithmetic operation is independent of the size of the numbers involved, then N can be factored in at most $O(\ln(N)^2)$ times that processing time.

I have a feeling that this might be a very stupid question, but, even allowing for the fact that in reality arithmetic takes longer for larger numbers, isn't $O(\ln(N)^2)$ still very much better than any known algorithm? I'm assuming that the numbers involved are at worst $O(N)$.