Re: Can EXPTIME and NP be separated via diagonalization?

Source: http://coding.derkeiler.com/Archive/General/comp.theory/2008−08/msg00019.html

- *From*: Zhu Guohun <ccghzhu@xxxxxxxxxxxxxxxxxxx>
- *Date*: Tue, 5 Aug 2008 19:21:04 −0700 (PDT)

On 8Eå, H10ö38 , tc...@xxxxxxxxxxxxx wrote:

> In article
> <4789ed7d−31a2−45d3−9783−ad37dcb6c...@xxxxxxxxxxxxxxxxxxxxxxxxxxxxx>,
>
> <cplxp...@xxxxxxxxx> wrote:
>
> > For example, suppose I could prove that if P = NP, there is an
> > algorithm that finds an algorithm that solves SAT.
>
> We already know how to write down explicitly an algorithm with the property
> that, if P = NP, it solves SAT in polynomial time. Just list all polytime
> algorithms and multitask over them all.
>
> In more detail, let A_1, A_2, A_3, ..., be a list of all polytime
> algorithms. Draw the following grid:
>
> A_1: step 1, step 2, step 3, step 4, ...
> A_2: step 1, step 2, step 3, step 4, ...
> A_3: step 1, step 2, step 3, step 4, ...
> ...
>
> To multitask over all these algorithms, just zigzag your way through this
> grid, as in the proof that the rational numbers are countable. If P = NP,
> then some specific algorithm−−−say A_n−−−will solve SAT in polynomial
> time. The multitasking algorithm will basically simulate A_n, while
> wasting only a polynomial amount of time trying other algorithms.

Let me guess your idea,
Suppose there is a CNF (x1 or x2 or x3) and (−x1 or x3 or x4)
Then the multitasks are follows

A1 step1 step2 step3
A2 −step1 step3 step4

If the A2 can be excuted, then the CNF is SAT.

But the algorithms A_i must " be modified" so to received the state
from A_j (i!=j), it is not the original algorithm.

Maybe there are more better reducaction approach from the mutitasks
and beyond my knowledge without to modify the original algorithm.

––––––––––––––––––

Zhu

.