Subject: memory vs cpu

Posted by aetherlux on Wed, 12 Nov 2008 12:15:58 GMT

View Forum Message <> Reply to Message

Hi,

I have detected a feature??? for all my programs written in IDL and I would like to have some feedback about if it is a possible problem with the code, hardware, etc.

My usual configurations are:

- \* idl 6.0; Debian GNU/Linux; core2 Quad 2.4; 4GB RAM
- \* idl 6.0; Debian GNU/Linux; core 2 Duo 2.0; 2GB RAM

99.9% of the programs that I run suck 99% of the CPU and 0.1-0.2%. An example of a typical top output:

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+

COMMAND

13964 juan 25 0 50296 5640 3432 R 100 0.2 21481:08 idl

My tipical code has several do and if loops, in some cases big arrays, and sometimes a hard work reading from files and written new ones. I usually include basic mathematical operations with floatings.

I have basic knowledge of programming and I would like to know possible causes and if you think that it is possible to solve/improve it.

Subject: Re: memory vs cpu

Posted by Craig Markwardt on Thu, 13 Nov 2008 04:25:00 GMT

View Forum Message <> Reply to Message

On Nov 12, 7:15 am, aetherlux <aether...@gmail.com> wrote:

- > Hi,
- >
- > I have detected a feature??? for all my programs written in IDL and I
- > would like to have some feedback about if it is a possible problem
- > with the code, hardware, etc.
- >
- > My usual configurations are:
- > \* idl 6.0; Debian GNU/Linux; core2 Quad 2.4; 4GB RAM
- > \* idl 6.0; Debian GNU/Linux; core 2 Duo 2.0; 2GB RAM
- > 99.9% of the programs that I run suck 99% of the CPU and 0.1-0.2%. An
- > example of a typical top output:
- >

- > PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+
- > COMMAND
- > 13964 juan 25 0 50296 5640 3432 R 100 0.2 21481:08 idl

>

- > My tipical code has several do and if loops, in some cases big arrays,
- > and sometimes a hard work reading from files and written new ones. I
- > usually include basic mathematical operations with floatings.

>

- > I have basic knowledge of programming and I would like to know
- > possible causes and if you think that it is possible to solve/improve
- > it.

If you put 'WAIT, 1.0' at the end of your FOR loop, the percent CPU utilization will be a lot lower :-)

More seriously, you will need to learn how to vectorize your code. The first things to vectorize are the innermost loops.

Craig