

目的

Linuxにて、複数のコマンドを並列実行したい。ただし同時実行数を制限したい。

きっかけ

私は研究の仕事をしていて、「パラメータを変えて複数の状況で計算をさせる」という実験を行うことが多い。そうすると、複数コアがあるマシンで並列で動かしたい。

以前は、コア数に応じて手作業で実験パターン数を分け、そのスクリプトを作っていたのだが、流石に面倒になってきて、自動化しようと決心したのであった。

最初に知った方法

とりあえず、waitというコマンドで「&を付けてバックグラウンド実行したコマンドが終わるまで待つ」が実現できるとわかった。

- (sleep 3 & sleep 2 & sleep 1); echo END
→3秒待ってから「END」が表示されるわけではない
- (sleep 3 & sleep 2 & sleep 1 & wait); echo END
→3秒待ってから「END」が表示される

なので、実験を動かすたびに変数の値をインクリメントして、その数が所定の数になるまで動かすとかをやっていた。

つい最近知った方法

xargs を使ってカジュアルに並列処理 - たごもりすメモ

<http://tagomoris.hatenablog.com/entry/20110513/1305267021>

- 普通、xargsは「標準入力から受け取った値をコマンドの引数にまとめて渡す」
seq 1 3 | xargs echo # "echo 1 2 3" と同等
- -Lオプションを付けると、標準入力から受け取った値を何行ずつ区切って渡すかを指定できる(それらは個別に実行される)
seq 1 3 | xargs -L 1 echo # "echo 1; echo 2; echo 3" と同等
seq 1 3 | xargs -L 1 sleep # 6秒待つ(1+2+3なので)
- -Pオプションを付けると、その個数上限で並列実行する
seq 1 3 | xargs -L 1 -P 3 sleep
待つのは3秒だけ(1秒待つ・2秒待つ・3秒待つのを並列実行している)

さらに芋づる式に知った方法

コマンドを並列に実行するGNU parallelがとても便利 - りんごがでている

<http://bicycle1885.hatenablog.com/entry/2014/08/10/143612>

コマンドの置き換えとかもできて便利

-j 1 だとただの逐次実行

```
$ seq 4 3 10 | parallel -j 1 'echo sleep {} start; sleep {}; echo sleep {} end: `date`'  
sleep 4 start  
sleep 4 end: 2017年 10月 7日 土曜日 13:54:59 JST  
sleep 7 start  
sleep 7 end: 2017年 10月 7日 土曜日 13:55:07 JST  
sleep 10 start  
sleep 10 end: 2017年 10月 7日 土曜日 13:55:17 JST
```

-j 3 で3つすべてを並列に実行

```
$ seq 4 3 10 | parallel -j 3 'echo sleep {} start; sleep {}; echo sleep {} end: `date`'  
sleep 4 start  
sleep 4 end: 2017年 10月 7日 土曜日 13:54:23 JST  
sleep 7 start  
sleep 7 end: 2017年 10月 7日 土曜日 13:54:26 JST  
sleep 10 start  
sleep 10 end: 2017年 10月 7日 土曜日 13:54:29 JST
```

-j 2 だと、「4秒待つ」と「7秒待つ」のどちらか(=前者)が終わってから、「10秒待つ」が実行される

```
$ seq 4 3 10 | parallel -j 2 'echo sleep {} start; sleep {}; echo sleep {} end: `date`'  
sleep 4 start  
sleep 4 end: 2017年 10月 7日 土曜日 13:54:39 JST  
sleep 7 start  
sleep 7 end: 2017年 10月 7日 土曜日 13:54:42 JST  
sleep 10 start  
sleep 10 end: 2017年 10月 7日 土曜日 13:54:49 JST
```

括弧やリダイレクトがそのまま書けるのもポイント高いところ

```
$ seq 4 3 10 | parallel -j 1 '(echo sleep {} start; sleep {}; echo sleep {} end: `date`) > log{}.txt'
```

参考になる記事

<https://siguniang.wordpress.com/2012/09/09/notes-on-gnu-parallel-the-command-line-power-tool/>

余談

終了までの時間を早くしたければ、基本的には「実行時間が長いものから実行する」のがよい。ただ、完全に最適な実行時間にしたいければ、全組み合わせを試すのに相当する程度の時間はかかる。

- 同時並列数を変えない場合、(すべての並び順を考慮して)終了までの時間を最適化した場合の時間と、単に「実行時間が長いものから実行する」とした場合での終了までの時間を比べると、後者は前者の $(1 + 1/3)$ 倍以内で収まることが知られている。

Graham, R. L. (1969). Bounds on multiprocessing timing anomalies. *SIAM journal on Applied Mathematics*, 17(2), 416-429.

おわりに

シェルスクリプトの「&」をもっとうまく使えるようになりたい。