

ネットワークプログラミング

Jacques Garrigue, 2005 年 1 月 25 日

5 分散計算

今までの例では、各クライアントがユーザーに使われることを仮定していた。通信の目的は異なる場所にいるユーザーをつなぐことであった。

それとは別に、計算の高速化を目的とした通信もある。コンピュータを計算資源だと考え、多くのコンピュータを束ねて計算を行う。

様々な構成が考えられる。

流作業 処理の各ステップを異なるマシンにやらせる。ただし、ステップの間に流れるデータが多ければ、高速な通信が必要になる。ネットワークの構造に関して考えなければならない。

データ中心 元々各マシンが異なるデータを保管していれば、各計算を必要なデータのある場所で行わせることによって仕事の分散と高速化ができる。それによって通信量が抑えられる。

独立計算 計算をうまく分けることによって、完全に独立に行える場合。依存関係がないので、どのマシンがどの計算をするかが自由になる。

多くの計算が同時に行われるとスケジューリングが重要な問題になる。特に流作業のような場合では、前のステップが終わらないと次が始められないし、逆に次のステップが滞っていると前のステップの出力を溜めていかないといけない。マシンのどれかが止まると全体に影響が出る。

それに比べて、計算が独立しているとスケジューリングが簡単になる。例えば、サーバを立てておいて、暇なマシンが仕事をもらいにいく。サボっても、結果が来なければもう一度別のマシンに頼めばいい。SETI@Homeなどがそういう構造をしている。

ちなみに、哲学者の晩餐は資源の配分によって計算が独立しなくなる例である。

実習

以下の式を分散計算によって高速化する．

$$\int_0^{10} \int_0^{10} \sin(x \exp(y)) dy dx$$

$$\frac{100}{N^2} \sum_{i=1}^N \sum_{j=1}^N \sin \left(10 \frac{i-0.5}{N} \exp \left(10 \frac{j-0.5}{N} \right) \right)$$

ここでは $N > 10000$.

- サーバは計算を $\frac{1}{100}$ 程度の量に分ける .
- クライアントが聞きに来ると、ある計算の範囲をもらう .
- その計算が終わったら、結果をサーバに返す .
- 途中で止められるクライアントがあるかも知れないので、分ける仕事が無くなれば、返事をもらっていない範囲を再び配る .