

人工知能システムをフラットなものと考えてみました。オブジェクト指向と専用MPU群による超並行処理の実現です。

コンピュータのサブルーチンコールはパラメータとプログラムアドレスを持って実行しますが、オブジェクト指向ではそうしたインターフェースもプログラム本体も一緒くたにして塊として、オブジェクトとして実行する相手に渡すことになります。だから、プログラムも自然言語の単語と同じく、コンセプトとして統合的に扱えるのです。

人工知能では、これらコンセプトの発火として、物事の処理を制御していきます。発火したコンセプトを登録する領野があって、発火したコンセプトがあるとそこに登録して、それを超並列処理環境をつくる専用MPU群がコンセプトの内容を見ながら処理して行くことになります。こうすると、発火とプロセス実行が独立して実現できます。しかも、プロセス実行を発火という処理によって制御できます。

発火は、ある領野オブジェクトにコンセプトを登録する処理で、小さなプロセスです。加算とか処理はいらない。設定命令があるだけで、アドレス処理もいらないかも。ただし、名前をアドレスに変換するサブシステムは必要で、そのシステムは超並行処理で行うことになるのですが。アドレス変換プロセスは加算とか乗除の機能が必要ですから、簡単なハードウェアMPUではないですが、発火処理よりは少ない回数行われるだけです（一度名前にアドレスを割り振れば、あとはアドレスを使えるから）、そんなに高度な並行処理はいらないわけで。

重み付投票システム（ニューロコンピュータ）が必要ですが、これも一つ一つコンセプトとして管理できます。そうして、計算はバックプロパゲーションの計算だけだから、加算と減算と比較演算くらいでしょうか。あとは発火に処理を任せる。これまた小さなMPUで実現できる超並行処理です。しかも、重み付を整数で行えますから浮動小数点演算回路はいらない。

あとは、基本的な処理はパターンマッチングによる、コンセプトの検索です。これは、有限オートマトンで実現できます。発火とニューロコンピュータよりは大きなプログラムを実行する必要がありますから、すこし負担が大きい。しかし、ニューロコンピュータよりは少ない処理だから、トータルではそんなに大きな負担にはならない。

そうして、色々な解析プロセスがあります。通常のMPUが必要になるでしょう。GPUでも良いかも知れませんが、作りこみの処理があるから、普通の汎用スーパーコンピュータの流用になるでしょうか。工夫すれば、回路素子が少ない汎用のアルゴリズムを見つけて、専用MPUにして超並行処理にすれば良いでしょう。

汎用スーパーコンピュータは誰でも目標のシステムを構築できるので、いろいろの発想

で利用して行くときのコストパフォーマンスが良いのですね。ですが、MPUは多くの回路を持つ必要があり、そんなに大規模な並行処理は実現できないという難点があります。汎用システムより、目標に特化したほうがより大規模な並行処理が実現できるということです。ただ、人工知能に目標を限れば、必要な回路素子は少なくできるわけで、また、人工知能技術の応用分野が大きくなれば、大きくならなくても大規模に売れるシステムに成長すれば、コストパフォーマンスの向上は期待できます。だから、人工知能というものの基本アーキテクチャを考えだすことには、大きな意義があると考えます。

おわり