

(1) ICOT研究開発総合報告

(発表者：ICOT研究所 総括担当次長 黒住 恭司
ICOT研究所 研究担当次長 古川 康一
ICOT研究所 研究部長 内田 俊一)

質問：アムステルダムから参りましたデンブリーと申します。今、お話のあった成功点について、たいへん興味深く拝聴しました。できま
すならば内田先生から、うまく行かなかった点
などもご紹介願えますでしょうか。

回答：失敗はすぐ忘れてしまうものですが、
ハードウェア研究に関しては当初は別の候補が
ありました。これは、知識ベースマシンと呼ば
れるものです。理論的には2種類の並列処理が
あります。1つはアルゴリズムプログラムで、
先ほど古川さんがおっしゃったようなものです。
もう1つはたいへん大きなデータベースのオペ
レーションです。データベースオペレーション
は、言ってみれば一種の連想型オペレーション
と考えられます。そして当初はそれら2つの課
題を設定しました。知識ベースマシンに関して
は、私どもはハードウェアの方に関心が向いて
いました。ところが、ハードウェアシステムを
作成するのは、ソフトウェアシステムを作成す
るよりかなり難しいんですね。そこで中盤に
入って研究の方向をちょっと変更して、知識処
理と並列推論マシンのアプローチを統合しまし
た。

そしてもう一つ、当初の研究では、もっとユニ
ークで興味深いアーキテクチャのアプローチ
を考えていました。例えば、非常に高度な知的
CADシステムを使って、データフローマシ
ンのアーキテクチャをVLSIチップにフルに実現
できるのではないかと思ったのです。しかしな

がら、半導体の技術に関しては、現在のPIM
のハードウェアを完成するに留まったわけです。
それは、データフローモデルを組み込んだハー
ドウェアよりは簡単なものです。つまり、この
プロジェクトの初期の段階において、ユニーク
なアーキテクチャの開発にそれほど重きを置か
ないことに決めていたわけです。というのは、
そのようなアーキテクチャというのは、ソフト
ウェアの研究とはかけ離れがちだからです。そ
こで、ハードウェアの研究者に対しては、ソフト
ウェアの研究者が好むようなハードウェアを
考えてほしいということをお願いしたわけです。
以上が2つの例です。

座長：古川さん、この質問に対して何かコメ
ントがございますか。

古川：私自身も後で失敗だったと思った経験
があります。それは、非常に野心的な知識表現
法で、マンダラと呼ばれるものです。アイデア
自体は良かったと思うんですが、ちょっと時期
尚早だったのではないのでしょうか。目標として
は、並列論理言語の上に非常に強力な知識表現
言語を作りだそうということでした。この種の
研究は現在でもまだ難しいと言わざるを得ませ
ん。しかし、可能性は大きくなりつつあると言
えるのではないのでしょうか。

質問：ドイツのレーベンシュタットから参り
ました。このプロジェクトは、マシンとソフト
ウェアを合体して1つのものにするというアイ
デアにおいてユニークなものであると思います。

先ほど内田さんは、ソフトを市販の標準的なマシンに移植できるとおっしゃいましたが、移植によってどのくらい性能が落ちるとお考えになりますか。先ほどのお話はPIMのデータだけだったわけで、PIM以外のマシンを使用した際の性能のデータはなかったようです。この種のソフトに、PIMマシンがどれくらい必要不可欠なのかをお話し願えますでしょうか。

内田：KLI言語のプロセッサを従来のマシン上に移植した場合に、どれくらい性能が落ちるかの評価はまだ終わっていません。移植を成功させるために解決しなければならない問題があるので、近山さんが今その調査を行っているところです。その問題のいくつかは、タグ・アーキテクチャのサポートに関わるものです。タグ・アーキテクチャのサポートは非常に単純なものです。並行ガーベッジ・コレクション、それにベクトル構造の遅延コピーや遅延アクセスを実現するためには非常に効果があります。もちろんご存知だと思いますが、関数型言語や論理型言語では、大きな構造のコピーは常に問題でした。KLI言語でのインプリメンテーションでは、コピーは実際に値が要求されて始めて行うということになっています。タグ・アーキテクチャのサポートは、この種のサポートには非常に本質的なものであると言えます。それで、多くの人がこのようなベクトル構造を使い出しています。これが1つの問題です。つまり、32ビットマシンの2ワードで1つのセルを表現する型を作成しなければなりません。これをPIMにインプリメントすると40ビットが1ワードになります。もう1つ問題となるのは、プロセス切り替えの制御をUNIX環境にどのように合わせるかということです。もちろん、UNIX環境では、きめの細かい並列処理というものを想定していませんので、UNIXのカーネルの一部を他のメカニズムで置き換えなければなりません。それで、性能の低下の評価はそん

なに簡単ではありません。完璧な評価結果というわけではありませんが、6か月ぐらいお待ち頂いて、この評価を要約した研究者の論文をご覧になっていただきたいと思います。

質問：ブラウン大学のピーター・ウェグナーと申しますが、まず最初にこのプロジェクトの達成に感銘を受けたことを申し上げたいと思います。1つのプロジェクトで、世界に大きな影響を与えたと思います。まず、古川さんに質問があるのですが、FGHCの探索パラダイムでは、探索が完了したことをどのように認識するのでしょうか。対象とするすべての問題について、完了を認識できるとは思えないのですが。完了したこと、または完了間際であることを認識できると保証できるような問題のクラスがあるのでしょうか。すなわち、探索の完了というのはどういった方法、または敷居値で評価できるのでしょうか。

古川：おっしゃる通りです。限定された問題のクラスでしか使えません。これは値域制限と呼ばれているもので、つまり帰結にでてくる変数は、すべて前提に出ていなければならないという制限です。したがって、この条件によって、ボトムアップ型の計算によって全解を求めることが保証されます。ボトムアップ型の計算を実行すると、その結果は基底項になり、帰結部に導入される追加の変数はありません。これが制限になります。主に複雑な数学の定理の問題ですが、その制限では表現ができない問題があります。しかしながら、診断や設計などの多くのアプリケーションには具体的なモデルがあり、値域限定を式に容易に定式化できると思われる。したがって、値域限定は実際にはそれほど重大な制限ではなく、有効であると思います。

座長：たいへん申し訳ございませんが、残念ながら時間が来てしまったようです。さらに、質問やコメントがお有りになる方は、明日のロビンソン教授の招待講演の後に時間を取れるの

ではないかと思います。また、皆様の中にはレセプションでお会いできる方もあると思いますが、そちらでよろしく願い致します。ありがとうございました。