
2) 第五世代コンピュータ 11年の成果と今後の課題

—— 今後2年間の後継プロジェクトの役割

ICOT・研究所所長 / 内田俊一

研究基盤化プロジェクト2年間の責任者となった内田所長は、かつて瀧所長の右腕としてプロジェクトを支えてきた1人であるが、ここでは「総合報告」として、過去11年間の研究開発成果を詳細に報告し、さらに次の2年間のプロジェクトで何を成すべきかについて語っている。その内容は、技術的な成果報告になっており、まさに第五世代コンピュータ・プロジェクトの研究総括と言えるものである。

タンポポの花のたとえ

ご紹介いただきました内田です。先ほどの 洲さんのお話にもありましたように、第五世代プロジェクト11年間の後を受けて、後継プロジェクトの2年間、洲さんの後をお引き受けすることにしました。

先ほど、洲さんが11年間の総括を非常によく整理してお話ししてくださいましたが、私はさらに具体的に11年間の成果を説明するとともに、2年間の後継プロジェクトをどのように進めていくかについてお話ししたいと思います。

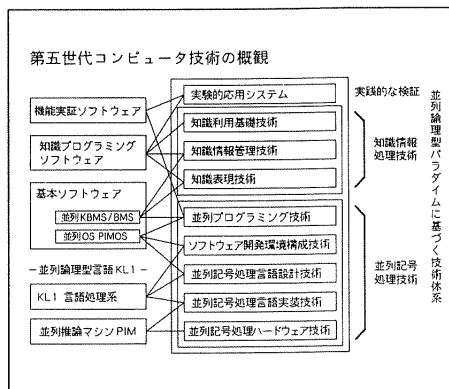
物にたとえてみるとわかりやすいと思いますので、タンポポの花のようなものを考えていただきたいと思います。最近、ICOTは、遺伝子の情報処理を手がけていまして、私もそのグループの立ち上げの段階で、それなりに生物の勉強をしました。タンポポの種から芽が出て花が咲くというところは、第五のプロジェクトで言いますと、当初の10年計画に対応すると思います。先ほど洲さんの話にもありましたように、最後に何とか花を咲かせることができたと考えているわけです。次は、その花をどうするかという話になります。生物、特にタンポポのようなものはよくできていまして、花は枯れますが、その後、その情報すべてを再び種の中に仕込んで、さらにその種が飛ぶように、綿毛のついた飛行装置という分配・拡散装置をつけて子孫を広めていくわけです。第五世代のプロジェクトも、その成果のでき上がった時点でバタッと終わってしまうと成果が世の中に広まっていかない、と最後のほうで思いまして、タンポポの綿毛ではありませんが、何とかその成果がより広く世の中に広まり、いろいろなところに辿り着

いて新たな芽を出す、そんな種子を作るようなプロセスをぜひ後ろにつけ加えたい、と思ったわけです。

それは、その前の段階で、私や若手研究者が自分たちの望みどおりの研究開発成果が得られたことを反映しているのかもしれませんが。ということで、これからの私の話は、咲いた花がどんなものであったのかということ、それから、遺伝情報を全部種子に詰め込んで、つまりわれわれで言いますと成果の情報を詰め込んで、どのように世の中に広めていくか、さらに広まった先で、どのように根つき、どのような花が咲くだろうかというような話としてお聞きいただきたいと思います。

第五世代コンピュータの成果を、いろいろ分析致しました(図1)。

図1 第五世代コンピュータ・プロジェクトの成果概要(1)



最終成果の報告

これは、皆さんのお手元にある、通産省の最終成果報告書の最初のページに書かれている絵です。第五世代の成果は、最後に第五世代コンピュータ・プロトタイプシステムと呼ばれるものにほとんどインテグレートされて

おります。そのシステムが、この図の左側でして、一番下に並列推論マシンPIMというのがあり、その上に並列論理型言語KL1があります。さらにその上に基本ソフトウェアとして、世界初の本格的な並列OSである、PIMOS、それからその一部をなす並列DBMSがあります。その上に知識ベース管理ソフトがくるわけですが、この部分は最後の出来上がりの具合いから、ほとんど知識プログラミング・ソフトウェアの一部として認識されています。この辺までは、しっかりツールとなるように作ってあります。それからこれをもとに、どのようにわれわれの社会にある知識を計算機の上に載せていくかという、知識処理の本来のテーマである知識プログラミングの諸技術、それを具現化したソフトウェアがあります。さらにその上、場合によってはその横かもしれませんが、さまざまな機能実証ソフトウェアを作ってきました。このようなものが、第五世代コンピュータのプロトタイプシステムという形でまとめられました。

この中に含まれている個々の要素技術がどのようなものであったかを、より広いコンピュータサイエンスのエリアから眺めてみてまとめたのが中央の部分です。

一番下にあるのが並列記号処理向きのハードウェア技術。ICOTが目指したコンピュータは、世の中で現在広く使われている大規模科学技術計算というような方向ではなく、より汎用的な事務処理一般を含めたものです。それは、聞きなれない言葉かもしれませんが、学問的には記号処理とひとまとめに呼ばれています。ICOTが目指したのは、記号処理向きのハードウェア技術といえます。それから、そのハードウェアの上にあるのが言語でして、このプログラミング言語というのが、コンピュータシステムではいつもおへそみたいなもので

中核を占めています。われわれの場合は、並列論理型という、並列処理の高級言語を採用しました。その言語の実装技術、また、その言語をどのように設計するかという言語仕様の作り方の技術、それからその言語を使う人にどうやって心地よい開発環境を提供するかといった言語関連のプログラミング環境を作る技術、この三つの技術を開発しました。さらにその技術および通常の並列処理の技術を用いて、問題をどのように定式化して並列処理のソフトを作っていくかというプログラミング技術、ここまですべてが主要な技術としてリストアップされています。この辺をひとまとめにして並列記号処理技術と分類できます。第五世代の技術は並列推論というキーワードで述べてきましたが、一般的なコンピュータサイエンスのエリアでは「並列記号処理技術」と呼ばれるものです。

一方、知識をどのように表現し、どのようにため込んで知識ベース化し、さらにそれをどうやって利用していくかという技術があります。こちらはまだ未発達で、並列記号処理技術のほうがかなり工学的完成度を高めているのに対して、こちらは依然として基礎研究の段階にあります。しかしながら、そこでつくられた個々の要素技術は、十分評価するに足る形にまとめられています。われわれはこの辺をひとまとめにして、「知識情報処理技術」と呼んでいます。この上にある実践的な検証システムというのは、一部はこの並列記号処理の技術で、もう一方は知識情報処理の技術の実践的な検証となっています。ここには、約二十数種の応用システムがあります。

この並列記号処理技術および知識情報処理技術のそれぞれについて、さらに展開して説明したいと思います。

まず、並列記号処理技術です（図2）。

図2 第五世代コンピュータ・プロジェクトの成果概要(2)

<p>1).並列記号処理技術（汎用的な大規模並列処理技術）</p> <p>-- 成果の概要 --</p> <p>知識処理、記号処理の高度並列化を達成</p> <p>→プロセッサ台数にほぼ比例した処理速度を実現 （500台規模で、従来の汎用マシンの50~100倍の高速化）</p> <p>→大規模な並列ソフトウェアの高い生産性を実現 （複雑な応用ソフトウェアが短期間で完成）</p> <p>→従来技術に欠けていた、汎用並列処理の技術を確立 本格的な汎用並列処理の時代の幕開けに先駆ける</p>

並列処理技術は、従来のコンピュータ技術の流れを見ると、科学技術計算の分野ではかなり大規模に使われてきました。このような大規模科学技術計算においては、計算の構造が予測可能なので、多数のコンピュータを使った場合の負荷配分などをかなり前もって準備して決めていけるということで、専門的にいうと、静的な最適化が図れるといわれていました。それに対して、記号処理、知識処理の分野では、困碁のプログラム、その他ゲームのプログラム等に見られるように、相手の打つ手によって次に何をやるかが非常に変わり得る。一般的に、AIの問題は、定理証明、その他を含めて探索問題に落ちるわけですが、探索の範囲が動的にコロコロ変わることから、並列処理の計算構造の組み立て予測が難しい。したがって、最適化が動的に行われなければならないという問題があるわけです。

この動的に行われていく部分を、プログラムとしてどう記述できるか、それからさらにOSのような管理ソフトがどのようにそれを効率よく管理していくかが、重要な問題として捉えられてきました。これが並列処理の問題の本質です。ここのところを解決したというのが第五世代プロジェクト、すなわちこの並列記号処理技術の成し遂げた大きなブレイク

スルーであるということができると思います。「高度並列化を達成」とシンプルに書いてありますが、ここが非常に大きな成果だと主張しているわけです。

説明が相前後しますが、従来からコンピュータの主要技術であった汎用大型機の技術が、ワークステーションやパソコンに徐々に地位を譲り渡し、分散処理の世界が構成され、その一方で、並列科学技術計算のほうはスーパーコンから、より汎用的な並列マシンに姿を変えつつあります。マーケットが、大規模並列、分散処理の時代になった。その中で、再び大きなデータベースサーバなどという形で、並列マシンがその中核に据えられ、新たなスーパーコンピュータとなる状況が起きているわけです。そのような汎用並列処理の時代がやってきているのに、大規模並列コンピュータをプログラムする技術は、必ずしも市場にないという現実が一方で起こっています。

第五世代の技術は、知識処理をするというシーズオリエンテッドなところからスタートしましたが、計算構造の予測不可能な問題を扱うということから、期せずして汎用的並列処理の技術を生み出すに至ったわけです。したがって、われわれの予測したところとは別の原因で、マーケットにおいて分散処理、並列処理がボトムアップに起こってきており、われわれの大規模汎用並列処理の技術が、そのようなニーズとぴったり合致して、さらに発展を続けるという可能性が見えてきています。こういう現実もとらえまして、後継プロジェクトでは、タンポポの綿毛を作るというような話が必要となってくることになります。

並列推進マシン「PIM」

次に、綿毛にする中身は何かを説明します。

記号処理、知識処理の技術は、並列処理を用いることですごく速くなるという人と、速くなるわけがないという人が相半ばしていました。しかしICOTは、500台規模で汎用マシンの50ないし100倍という、プロセッサ台数にほぼ比例した処理速度を実現できました。実際、この値は私のような中核に位置していた者にとっても、予想を上回るものでした。

また、定量的な評価は、なかなか難しいですが、KL1という言葉によるソフトウェアの作成効率がよいということがあります。並列推論マシンの規模が64台、128台、256台、512台と増えていくのに対して、ICOTの応用プログラムを書いていたチームは、約2週間ぐらゐの遅れでソフトウェアの拡張を行っていたことから見ても、KL1のソフトウェア生産性が高いというのを肌身で感じています。それがどのぐらい高いかというのが、今後いろいろなところで定量的に評価されるといいなと思っています。こゝうことで、従来技術に欠けていた、こゝうな汎用並列処理の技術を確認でき、本格的な並列処理の時代の幕開けに先駆けることができたこと、自画自賛ではあるかも知れませんが、総括しています。

こゝうな並列記号処理の部分について、さらに具体的に紹介していきたいと思ひます。詳細は、近山隆第一研究部長及び新田克巳第二研究部長、長谷川隆三次長等による、本日午後の講演で述べられます。私は、そのアウトラインだけ述べたいと思ひます。

まず、並列推論マシンPIMですが(図3)、これは5つのモデルをつくりました。

それぞれに名前がついていて、PIMモデル-p, -m, -cなどとなっています。これらは、それぞれ異なるアーキテクチャを持っていて、ハードウェアやアーキテクチャの研究要素がたくさん注ぎ込まれています。しかしながら、す

図3 研究開発成果(1)

1) 並列推論マシン PIM
PIM/p 512PE, PIM/m 256PE + 128PE, PIM/c 256PE など、5モデル
大規模モデルでは、100MLIPS以上の性能
知識処理の応用ソフトウェア研究ツールとして安定稼働
インターネット経由で国内、海外より利用可能

べてのモデルがKL1およびPIMOSという一つの言語、一つの環境をサポートするので、ユーザは自分のプログラムをPIM/pに載せたり、PIM/mに載せたりして、自分のプログラムの計算構造とアーキテクチャのマッチングの具合を見るような研究ができます。

性能的には、先程も申し上げましたが、現在のワークステーションが今一番速いマシンになっているわけですが、100MIPSクラスがあります。そういうワークステーション1台に比べて、50倍から100倍の性能を達成しています。ただ、こちらのほうは512台つき込んで100倍です。早い話が、C言語で書かれたプログラムを1台のワークステーションで実行すると、PIMのプロセッサを5台持ってきたのと、性能が同じぐらいであるということです。PIMのプログラムはKL1で書いてあるわけですが、そのオーバーヘッドが5倍以内で済んでいるということを示しているわけで、KL1で書くと、台数が100台、200台になっても追従できる。一方、Cで書かれたプログラムは、ワークステーション1台で終わり、拡張性はないわけ、並列処理の時代を迎えて、このKL1という言葉、およびこのPIMというマシンが非常に有効に働くことがおわかりいただけるかと思ひます。

このPIMは試作機ですが、担当いただいた各メーカーの工場で実装していただいたことも

あって、非常に安定して稼働し、応用ソフトウェアの開発ツールとしては十分な安定性を持っています。国際共同研究等もこの上で行われていて、インターネット経由で海外、国内から使用可能になっているという状況です。

ここで、写真をお見せします(図4)。ここにあるのがPIMモデル-pというマシンです。これは、最大構成512台、一つの筐体に32台のプロセッサが入っています。このモデルの特徴は、要素プロセッサを8台ごとにクラスタという単位にまとめていることです。したがって、512台で64個のクラスタがあるように見えます。個々のクラスタの中は、自動負荷分散という機能があり、プログラマは8台が1個の非常にパワフルなコンピュータであると見なして負荷配分をします。8台の中の細かな負荷配分は自動的に行われます。1筐体32プロセッサで、1列4筐体で128台、これが4列まで拡張でき、512プロセッサを構成しています。

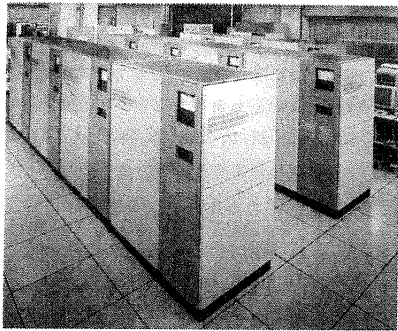


図4 512台の要素プロセッサからなる並列推論マシン PIMモデル-p (PIM/p)

これ(図5)は、PIMモデル-mと称するもので、こちらはさっきのクラスタのような構造を持っておらず、16×16の単純な碁盤の目の構造になっています。

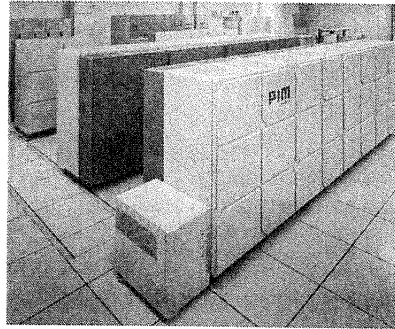


図5 256台の要素プロセッサからなる並列推論マシン PIMモデル-m (PIM/m)

PIM/mでは、ソフトウェアを書く人が256台の1個1個のプロセッサを意識し、負荷配分していきます。構造が単純なので、ソフトウェアのモデルを簡単に書いて、それをマシンの上に直接載せます。どこが良かったか悪かったかというマッピングのあり方を見るときも、単純であるがゆえに実験がしやすくなっているわけです。このモデルは、256PEが最大構成ですが、このモデルのほかに128プロセッサの小さいモデルも作っています。ここに示しているのは1筐体に32台、4筐体で128台、2列で256台です。

以上のようなハードウェアが、現在のICOT

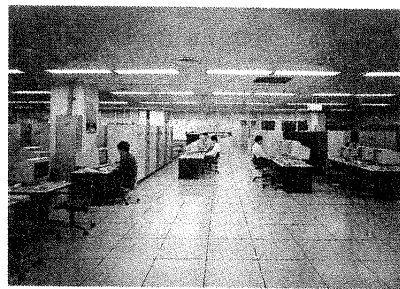


図6 稼働中の並列推論マシンPIM
向かって左側が、512台規模のPIMモデル-p(PIM/p)
右側が、256台規模のPIMモデル-m(PIM/m)

のコンピュータパワーを支えています。本日のデモでも、三田国際ビルの計算機室（図6、7）から、ネットワーク経由でPIMを使って、プレゼンテーションの合間にさりげなくリアルタイムデモが挿入されるという仕掛けになっています。



図7 ICOT マシンルームの並列推論マシン PIM
写真の手前に、PIMモデル-m (PIM/m)、右奥にPIMモデル-p (PIM/p)、さらに左の端末群の奥にPIMモデル-c (PIM/c) が見える

リアルタイムデモがあまりにもスムーズに挿入されると、アニメーションの絵が入ったのかなという感じで、あまりその辺の断続が明確でない分だけ、本物のデモだというリアリティに欠けるという問題がありますので、注意してご覧いただきたいと思います。

基本ソフトウェアと言語

さて、並列記号処理技術のもう一つのポイントである基本ソフトウェアと言語について、もう少し紹介したいと思います（図8）。

KL1は、これまでICOTがずっと開発を続けてきた言語ですが、この言語は並列処理の高級言語であり、同期メカニズムはデータフローと称するメカニズムによっています。したがって、ユーザはその同期の個々について関与する必要がありません。また、メモリについては、仮想的には無限大あるということ

図8 研究開発成果(2)

2) 並列論理型KL1と基本ソフトウェア	
KL1言語処理系	→分散データ管理、分散実行管理に多数の新技术
並列OS PIMOS	→分散管理により管理ボトルネックを解消 生産性の高い並列ソフトウェアの開発環境
並列DBMSKappa	→柔軟な非正規関係モデル分散並列処理による高速化

でプログラムが書けます。したがって、そこには裏方として分散ガーベージ・コレクションのメカニズムが組み込まれています。このような高級言語によって、初めてユーザは何百台ものマシンをプログラムする自由度を与えられるということになります。

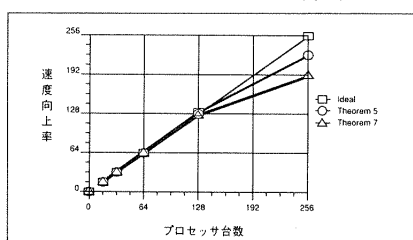
この言語の特徴のもう一つは、実際にアルゴリズムを記述する部分、および実際にできたプログラムを256台なり512台のマシンに割り当てる部分が、別々の二段構えにプログラムできることです。ユーザは、まずアルゴリズムを記述して、さらにその後、プラグマと称する指定をつけ加えることで、自分のプログラムのどこの処理を、どのプロセッサに割り振るかを指定します。第1段階では、通常アルゴリズムのチェックを行い、第2段階では、ランタイムモニター等を使って、プロセッサ上の負荷配分がどのように行われているかをモニタリングツールを見ながら調整し、最高効率を得るというパフォーマンス・デバッグを行います。この2段階のステップによって、プログラムができるようになっています。

並列OSのPIMOSには、ユーザが使うメモリやプロセッサ等の資源管理をできるだけ分散してやるようなメカニズムがうまく組み込まれており、それによって管理のボトルネックを解消しています。同時に、アルゴリズム

のデバッグや、パフォーマンスのデバッグを効率よく行う環境がこの上に載っています。そのほか、基本ソフトの一部として、並列のデータベース管理ソフトウェア Kappa が実装されています。例えば PIM モデル-m ですと、256 台のプロセッサとともに 80 ギガバイトのメインメモリがあり、これを一種のディスクキャッシュとして用いることで、非常に高速の検索が可能となります。このような並列データベースのアクセス、管理が実際に行えるようになってきているのが、この Kappa です。

では、次にこのような基本ソフトウェアと PIM の組合せによって、どのくらいの高速化が行われたかを紹介したいと思います (図 9)。

図 9 並列定理証明システム MGTP の高速化



先ほど、台数に比例した並列処理の速度向上という話をしました。これについては、どのくらい速くなるだろうかという議論が、ICOT 関係者の間で多々ありました。並列処理のソフトウェアをつくるときの一つのアナロジーとして、会社の組織などを思い浮かべるわけです。従業員が 100 人いるような会社を非常にうまく切り盛りしたとして、1 人のときに比べてどのくらい生産性が上がるだろうかというのが、並列処理の効率を考えるときの一つのアナロジーです。人間の会社と同じで、計算機の上で大きな一つの仕事を与えられると、

まずそれを分けるわけです。誰かの仕事が終われば、そこに新たな仕事を供給する。終わった仕事の結果の取りまとめに走り回るといような管理のオーバーヘッドまで含めると、100 人のうち 60 人ぐらいがフル稼働していればよくて、場合によっては 30% ぐらいでもいいかなと考えがちなわけです。

次に定理証明問題の速度向上についてお話します。定理証明という問題は、並列に処理できる多くの部分に分けられますが、大規模科学技術計算のような方程式がきちんとたっているわけではありません。したがって、動的に負荷の再配分を繰り返す必要があります。それを徹底して行ったということで、最終的にこのような良好な成果が得られたわけです。これは、256 台のプロセッサを使った例です。これですと、220 倍ぐらいの速度向上が得られているわけです。こういう結果に勇気づけられて、並列の CAD をやっている人や、法律の推論システムをつくっている人が、いろいろなノウハウをお互いに教え合い、プロセッサ台数にかなりよく比例した速度向上率が最近では得られるようになってきました。

したがって、記号処理というような問題は、そのなかでは計算構造が非常に非定型的で予測しにくいとはいえ、プログラミング言語や OS が徹底した動的な負荷配分やその他の最適化を図ることで、このような台数にほぼ比例した速度向上が得られることがかなり確信できるまでになってきました。このような確信が、後ほど申し上げる、より本質的な知識処理の問題へアタックするときの前提を、大幅に変えることとなります。

以上で、記号処理の部分を終わらして、知識処理の説明に移りたいと思います (図 10)。

図10 知識処理技術

-- 成果の概要 (1) --
従来技術より数歩先んじた汎用並列処理の応用を開発
→ 応用領域を、工学分野から、広く自然科学や、社会科学などの分野に
広げ、知識処理の本質的な問題へ踏み込む

知識処理へのアプローチ

まず、ICOTにおける知識処理へのアプローチですが、前期からいろいろな基礎研究を行ってきましたが、並列処理の技術が固まるにつれて、その前提が変わってきました。要するに、マシンが出来たし、並列記号処理のパワーも上がったから、何かかこのいい知識処理の応用はないかという発想が出てきたわけですね。もちろん、それ以前から知識処理の応用でエキスパートシステムなどを手がけている人はいたわけですが、それとは別に、1,000台規模のマシンが動くとき、そのマシンの上に載せるだけのボリュームのある応用がないということが、中期の終わりの頃から叫ばれ始めたわけですね。

考えてみますと、科学技術計算の応用分野には実際に並列マシンがありますから、どんどん応用の人が入ってきて大きい並列の応用プログラムが作られたのに対して、もともと従来型機は記号処理は不得意でしたから、大規模な並列応用の開発はなされていないわけですね。ということで、並列推論マシンをいっぱいにするような新しい応用を掘り起こそうという発想、つまり、自分たちで応用まで手がけようという発想が出てきたわけですね。

そういう発想からいろいろ調べた結果、工学の分野には、VLSI-CADなどの応用が結構あることがわかったわけですが、もう少しそれを広げようとして目についたのが、遺伝子

の情報処理でした。1988年ぐらいからヒトゲノムの解読計画がアメリカで開始され、非常に大量の遺伝子情報がデータベースに蓄えられるという状況が起こってきた。この遺伝子、つまり、DNAの記号列、もしくはタンパク質の記号列というのは、それがどのような意味、どのような構造を持つものであるかは、わからないにもかかわらず、その読み取り技術だけがどんどん進歩して、未解読の文字列が山のようにたまるという状況が起こっていたわけですね。これには、暗号解読のような処理を必要としますから、生物学サイドは記号処理のパワフルなマシンを求めていました。では、この辺をひとつ開拓してみようということで、われわれは生物学的な応用の研究を始めました。

さらにもう少し先へ行こうということで、法律のエキスパートシステムの研究を始めました。いろいろな判例、規則が山のようにあって、それらに基づいて新たな判断をすることですから、これも大規模な知識ベースを基本とした推論を必要とします。

このように応用の研究を進めると、同時に、より本質的な知識処理の問題に踏み込むことになります。この本質的な問題が何であるかは、また後でお話したいと思います。

ICOTの知識処理のもう一つの特徴として、

図11 成果の概要 (2)

述語論理に基づく知識表現の実用性を示し、種々の応用分野の知識ベース作成のための中核技術を確立
大規模並列処理のパワーを土台とした定理証明システム
演繹オブジェクト指向DB言語などの知識プログラミング言語知識処理、記号処理の典型的な応用例でそれを実証
→ 学際領域の研究者、研究分野育成が次の課題
知識情報処理技術の将来展望を提示

PIMやPIMOSの記号処理技術の評価用、評価の検証用という形で知識処理であったことがあげられると言いましたが、当然それ以前から、第五世代のプロジェクトには知識処理研究をやるうえでの大方針があったわけです。それは何かというと、今さら耳ダコでしょうが、述語論理、すなわち論理に基づいて知識を扱っていきこう、ということです(図11)。

論理に基づいて知識を扱うというのは、遠いギリシアの昔からなされていたのですが、コンピュータにのせて機械化する研究では不発でした。その一つの理由は、一般的に述語論理を証明していく手段、手続きというのは非常に手間がかかるもので、従来の計算機では処理速度が遅く、その論理自体をプログラミング言語として使うことができず、足踏みしていたのです。

ところが、第五世代プロジェクトでは、そういう知識処理の技術の中に並列処理のパワーを持ち込む前提でやってきました。その最初の突破口を開いたのが、まさにこの並列定理証明システムだったということができています。先ほどから、世界最高速とか、定理証明システムが台数に比例した処理速度向上を実現したと言いましたが、これは論理に基づく知識処理が、並列処理と結びつくことで実用レベルに達し得るということを宣言できたといえますか、明確にできたということです。これは、非常に大きなことで、これによって、論理に基づく知識表現、これは専門的には演繹的な知識表現とかディダクティブな言語と称するわけですが、そのような言語が実用になるとということです。そこで次は、知識を記述するというような側面から、その表現力がどうかというようなことに研究のポイントが移っていくわけです。

研究者は、当初から演繹的な言語にそれな

りの見通しを持って研究を行っていたわけですが、われわれのような研究を管理する立場からいうと、これがものになるかどうかの一つの条件が、その言語の効率的実装の可否にあったわけです。これでやっと条件が整ったといえるので、この辺をもっと本格的に研究しようということになるわけです。表現力を増すという立場からすると、現在いろいろなところで流行語にもなっていますオブジェクト指向に基づく表現、これを組み合わせようということで、研究が進められました。

さらに先ほど申しました記号処理の応用の中で、法律や遺伝子などの応用まで研究しますと、だんだん知識処理、記号処理のユーザー側からの要求が強くなってきます。中でも、こういう論理に基づく知識表現方式が有効であるということが、まだ傍証でしかないかもしれませんが、示すことができるようになってきます。この辺は、第五世代プロジェクトの成果としてはまだ要素技術レベルで、先ほどのKL1のプログラミング環境のように、これこれの知識表現言語でこのように書いていただくと、エンジンとして定理証明のプログラムが動いて、皆さんは並列処理の中がどうなっているかは知らなくても、論理をベースにして書かれた知識表現がプログラムとして動いてしまいますよ、とは、まだいえないわけですが、そういう方向が現実的であることを示し得たといえます。そうすると、将来、コンピュータと生物の、あるいは法律とコンピュータの両方にまたがる分野などの新たな学際的な研究分野が生まれ、その立ち上げが、面白く、なおかつ難しい問題として登場してくることが予想されるわけです。

次に、個々の課題の説明に移りますが、この辺は午後にも説明しますから、簡単にしたいと思います(図12)。

図 12 主要研究開発成果(1)

1) 知識プログラミング技術
並列定理証明システム (MGTPシステム) →世界最高速の定理証明システム 高水準の推論エンジンとして利用可能
演繹オブジェクト指向DB言語と知識ベース管理システム (Quixoteシステム) →述語論理に基づく知識の記述 仮説推論の記述など豊富な機能
並列制約処理言語システム (GDCCシステム) →制約記述でプログラミングが容易。並列処理による高速化

知識プログラミング技術の将来の目標は、こういう知識表現言語ができましたから、これでプログラムを書いて下さい、皆さんのお手元にあるいろいろな知識、お医者さんだったら、カルテの中書き込むデータとか、弁護士さんだったら新しい判例とか、そういうものをどんどんお書き下さい、と言えるようにしたいわけです。こうして、マッキントッシュやパソコン上のLotusのような感覚で知識をどんどん書いていくと、皆さんのパーソナルな知識ベースが出来て、新たな質問をすると、それ相応のインテリジェントな答えが返ってくるというような知識プログラミング、知識処理の技術というのをつくりたいわけです。しかし、そのまだはるか手前にいるわけで、ここでは要素技術についての説明をします。

エンジンとして見た場合の定理証明システム、これは今言ったような論理を根幹とする言語を実用化する非常に重要なポイントとしてとらえることができます。もちろん、定理証明の研究者にとっては、これ自身が非常に強力なツールであります。これについては、きょうの最後に長谷川次長から詳細が説明されると思います。

知識表現に関しては、演繹オブジェクト指向のデータベースの言語、およびその管理シ

ステムとしてQuixoteが研究されています。これは、述語論理をもとにして、その上にオブジェクト指向の殻をかぶせ、いろいろな形で知識がうまく表せるように、現在の仕様は非常に欲張った形になっています。非正規形の関係演算も扱えるような、非常に実用的な側面も加味した言語になっています。しかしながら、言語仕様を欲張っただけに、実行速度が伴っていないので、処理系自体の研究がさらに必要であるという状況です。

もう一つの高級なプログラミング方法として、制約で記述する手法があります。これは、制約解消系ソルバーなるものが前もって準備され、それが特定のドメインの知識を持っているということで、ユーザはそのソルバーに不足する分の知識、もしくはソルバーが持っている知識間の制約関係だけを書き足すことでプログラムが書けるというのが理想ですが、そこにも非常に多くの計算量が必要になります。一般に、人間が楽をしようと思うと、その何倍もの計算量が必要になるわけですが、これもその類でありまして、それについてはGDCCと呼ばれる並列の制約処理言語およびそのシステムが出来ています。こういうものが、知識をプログラミングするための、数は少ないが非常に重要な要素技術として確立してきています。

機能実証ソフトウェアとしては、第五世代プロジェクトの後期に、先ほどの知識表現言語等と並行して研究を進めてまいりました(図13)。

両方のグループは積極的に情報を交換しており、ユーザがどのような知識表現言語を望んでいるかを、知識表現言語のグループはよく勉強していますし、逆に応用の研究者たちは、ICOTの中にある使えるものは何でも使おうという精神で、いろいろプログラムを書い

図 13 主要研究開発成果(2)

<p>2) 機能実証ソフトウェア (知識処理、並列記号処理の典型的応用例)</p> <p>法的推論システム ルールベース推論と事例ベース推論の組合せ(HELIC-II) → 法令の論理的解釈と知的な判例探索により被告の判決を推論 遺伝子情報処理システム → 蛋白質やDNAの配列解析の並列化 モチーフ抽出、蛋白質の立体構造の予測</p> <p>LSI-CAD → LSIの配置・配線・論理シミュレーション 配列化のアルゴリズムを工夫して高速化 その他、自然言語処理システム、囲碁システム、プログラム生成システムなど。</p>
--

ています。

機能実証ソフトウェアの内容を少しずつ説明しますと、まず、ICOTで今一番インテリジェントなソフトだろうと思われる法的推論システム。これはHELIC-IIという名前で、現在、刑法の問題を扱っています。約100ぐらいの判例が知識ベースとして入力されており、さらに刑法の関連の諸法規が入っています。この法規と判例を、ルールベース推論エンジン、事例ベース推論エンジンを用いて検索し、ある事件が起こった場合、その被告にどのような判決が与えられるかを推論しようとするものです。これも、並列推論マシンの上で実行しないと、答えが出てくるまでに数時間かかるというようなオーダーなので、並列推論マシンのパワーが有効に働いています。ちなみに、ルールベースの推論エンジンは、並列定理証明システムMGTPをそのまま使っています。

遺伝子情報処理については、初期の頃は、コンピュータ・サイエンス側の研究者が生物の技術用語を学ぶのに精いっぱいという時代が1年ほどありました。最近ではかなり中身もよくわかり、DNAの配列やタンパク質の配列の

解析を研究しています。さらに、その中から、非常に意味のある配列であることがすでにわかっている部分、それをモチーフと呼んでいますが、そういうものをうまく引っかけて取り出してくる、あるいは、さらにタンパク質の立体構造を予測して、その性質を類推しようというようなレベルに研究が進展しています。これも今日の午後、もしくは明日の各論の発表で、デモを交えて皆さんにお目にかかれると思います。

それから、LSIのCADですが、これは必ずしもICOTのマシンやソフトウェアを知識処理のマシンとして使っているわけではありません。非常にローレベルの並列マシンとして、かなり無理に使おうというような側面がありますが、KL1でLSIの配置、配線や論理シミュレーションを書いています。そうすると、ゲートの動作や、個々のロジックの1が0に変化するというようなイベントを扱うわけですから、1個1個の処理単位が非常に細かい。そういう細かいものを推論マシンのような要素プロセッサに割り振って並列に実行しようとするわけですから、荷物を大きい包みで送ったけれども、中は1円しか入っていなかったというような問題が起きる。つまり、パッケージングのコストが大きく、通信コストと、中での処理とのバランスがなかなかとりにくいのです。

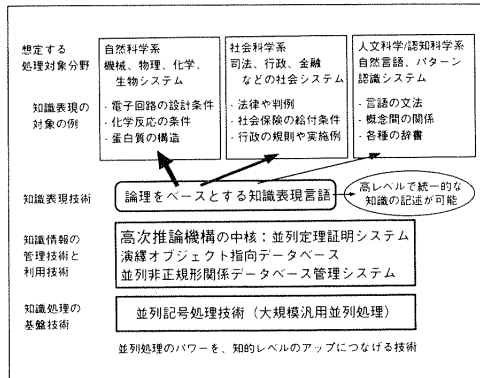
しかしながら、アルゴリズム等を工夫することで、かなり高速なものができています。実際、各メーカー、その他で並列のCADシステムを作るという話は昔からあったわけですが、実際に汎用の並列マシンがあったわけではないので、プログラムとしてはあまり研究されていませんでした。それで、現在メーカーでCAD等を研究している皆さんからは、非常に興味を持って見ていただいています。これからUNIX系の並列マシンが出れば、こういうもの

はすぐに役に立っていくのではないかと思います。

そのほか、自然言語システム、囲碁システム、プログラムの仕様をロジックで書くと、定理証明のようなプログラムが動いて並列プログラムが自動的に生成されるシステムなど、興味深いシステムが多々作られました。今日は、そのあたりの説明は省略致します。

以上のように知識処理、知識表現のための

図 14 第五世代コンピュータ技術の近未来展開



言語やエンジン、それから応用の一端の研究を進めてきた感じから、将来の知識処理の方向というのはこうなるだろうという簡単な絵が描けてくるわけです（図 14）。

近未来展開のイメージ

ここでは、かなり先の時代の絵を書いていますので、PIM、並列処理の管理等は一つの箱の中に入っています、いってみれば、1個の非常に強力なエンジンがあるというような単純化した絵になっています。この上に、先ほど説明した演繹オブジェクト指向の言語や、定理証明のエンジンがはまりますと、それらがうまく融合します。実際にはまだですが、将

来には融合するとしますと、この上に論理をベースとする知識表現言語がくる。この言語は、従来のワークステーションの上でのエキスパートシステムの知識記述用言語と違って、特定の分野だけでなく、いろいろな分野に使えるだろうということが予測できます。自然科学系の分野ではうまく知識が記述できますが、社会科学系分野では難しい。

自然科学系の場合には、例えば電気的な現象だとマックスウェルの方程式とか、キルヒホフの法則とか、非常にマクロなビヘイビアが単純な式でまとめられた、いわば神の声のようなものが定式化されています。それを見ながら部分的な知識をプログラムしていくと、部分的な知識をどんどん拡張していったときに矛盾が起きにくいのではないかなと思える節があります。したがって、このような言語で書いていくと、何となくうまく書いていくような気がしますが、社会科学系になると、どうも大変である。なぜかという、人間の混沌とした社会で、非常に頭のいい専門家がコンシステントな法律等をつくるといっても、自然科学におけるきちんとした体系のようなわけにはいかない。上位法、下位法等いろいろな法律があるわけですが、それが数学的な意味できちんと集合の包摂関係に合っているかという、そんなことはない、例外だらけだということで、扱いが難しい。

さらに一歩進むと、ICOTでは、残念ながら今年で一応終息させていますが、囲碁システムの研究もしてきました。囲碁の参考書はいろいろあるわけですが、あれは人間をコンピュータとして記述した知識であって、計算機が論理を扱えるようになったからといって、一足飛びに論理の上にその知識を展開するというわけにはいきません。「左右同形、中央に手にあり」というような非常に感覚的な説明を

どうやってこの上に載せるかというレベルまで到達するには、まだまだ距離があるような気がします。

ということで、知識情報処理のアタックすべきところは、もし計算機の上にそういう知識を載せたいのであれば、論理というものを使えるんだという前提に立って、その知識自体をもう一回再構成する必要があるという点です。この分野の研究は、もはやコンピュータ・サイエンスの枠だけにはとどまっていません。それぞれの分野の学際的な研究者を育てながら研究していかなくてはいけないということ、こういう絵を見ながら非常に切実に感じます。第五世代の成果というのは、そのような場合に、知識をいろいろな形で定式化し、分野ごとの難しさ等を比べる上での共通のテストベッドになり得るということ、ここでは言いたいわけです。このような考え方に基づくと、こういうようなものを新たな知識処理研究の研究基盤とするという基盤化プロジェクトの目標を言いたくなるわけです。

「研究基盤化プロジェクト」

ということで、研究基盤化プロジェクトについてお話ししたいと思います（図15）。

これは計画ですので、まだそんなに詳細に入ることはできません。しかしながら、並列処理というものがマーケットの中で主要な技術として使われるであろうことが非常に顕著になってきました。第五世代が始まった頃は、並列処理は一步も二歩も先に進んだ夢の技術で、市場技術とのリンクはあるとしてもずっと先のことだと半ば言い聞かせながら研究を進めてきたわけです。プロジェクトの終わりに際しても、市場技術とのオーバーラップはないのか、ということを考えていましたが、最

図15 第五世代コンピュータの研究基盤化プロジェクト（5G後継プロジェクト）

－研究開発目標－

第五世代コンピュータ技術を既存コンピュータ技術と融合させ、これから本格化する並列処理や知識処理の技術開発の研究基盤とする。

1. Unixベースの並列マシン上で動作するKL1処理系（KLIC）を開発しPIMOSなどの基本ソフトウェアを移植
2. 知識プログラミング技術や実験的応用システムを開発開発 これらをICOTフリーソフトウェア（IFS）として広く普及 ※すでに、海外、国内あわせて約1,100人に、のべ6,400本のソフトウェアを配付（インターネット経由のFTP利用）

後の最後で、ご存じのように市場のほうが本格的な並列技術に移る兆候が見えてきて、UNIXなども並列のプリミティブを一生懸命取り入れる時代になってきた。それで、第五世代技術をタンポポの綿毛の種に仕込む。仕込めば、綿毛の種が朽ち果てる以前に、どこか寄りつくしが見出せようだという気がして、基盤化プロジェクトをスタートさせようということになったわけです。

それと同時に、先ほど説明した知識処理の技術。これはもはやコンピュータ・サイエンスの枠の中では発展する余地が、ないわけではないが、あまり残されていない。ですから、UNIX系のマシンの上に、たとえばKL1を載せて、いろいろなところに種として飛ばせば、もしかすると法律の専門家で並列処理の好きな人がいて、それが自分のマシンの上にその種をとらえて花を咲かせてくれるのではないか、というような期待が広まってくるわけです。というわけで、後継プロジェクトでは、UNIXベースの並列マシンを対象として、その上でKL1やPIMOSが動くようにしようということが一つの作業目標となっています。

もう一つの作業は、知識プログラミングの技術や、応用システムの研究開発をさらに進めて、2年間経った後、綿毛として飛ばすとき

の綿毛のクオリティを上げておいてやろうということ。後継プロジェクトで作ったソフトウェアは、ICOTフリーソフトウェアということで無償公開する前提で進めています。ちなみに、現在まで第五世代プロジェクトで開発された77本のソフトウェアが、アノニマスFTP、インターネット経由で皆さんに持っていられています。4月末時点で約1,100人が6,400本をコピーして持っていった。ということなので、大変たくさんの人が持っていられているわけです。

一方で、持って行ってどうしているんだろうというのが非常に気になります。というのは、このKL1で書かれたソフトは、PIMの上でないと動かないわけですから、いくらソースといっても、持って行って単に読むしかないわけです。それもあって、例えばKLICのようなKL1の処理系が市販のパラレルマシンの上で動くようになると、ここで持って行って読んでおいたソフトがちょうど動くようになる。まだ後継プロジェクトの中身がそれほど広まっているとは思えないので、それをあてにして、飽きないうちに早くKLICを提供して、実際に走る環境を持ってもらいたいと思っています。

この図(図16)は後継プロジェクトのプロフィールです。

研究開発の期間は2年間で、今年の4月から走り始めています。初年度予算は約14億円で、第五世代の本体が花を咲かせていた時代に比べると、非常にわずかです。プロジェクトの目的は、先ほど言いましたように綿毛の種をつくることですから、このぐらいで良いということで進めています。体制としては、従来は7つの研究室に分かれて、約90人近い研究員がいましたが、それは大分減って現在約40人。それに加えてソフトウェアの開

図16 プロジェクトのプロフィール

1.研究開発期間	平成5年4月から2年間
2.研究予算	平成5年度 約14億円
3.ICOTの体制	ICOT組織 研究所→2研究所+研究計画部 事務局→総務部+調査国際部 研究員 研究所 約40人+ソフトウェア開発要員 中核研究ツール PIM/p 512PE、PIM/m 256+64+32PE PSI-III,SS10などワークステーション 市販の並列マシン (AXP7000,Dragonなどを予定) (国内、海外へネットワーク接続)

発要員として、大体40人ぐらいに協力してもらっています。中核のツールは、PIMのシリーズと、PSI-IIIを約100台、それにワークステーション、それから、市販の並列マシンとしてDEC社のスーパーサーバAXP7000とサン・マイクロのDragon等の設置を予定しています。すでに入っているものもあります。これらすべてのマシンは、国内、海外にネットワークで接続し、アクセス可能です。

研究開発内容

研究開発内容については、既存技術との融合ということから、フォーマルなテーマは「融合型推論技術」という表現になっています(図17)。

図17 研究開発内容

融合型基本ソフトウェア技術	
1.汎用マシン上のKL1処理系 (KLIC)	'93年9月逐次版KLIC初版、'94年4月並列版KLIC初版
2.並列OS PIMOSや並列DBMS Kappaの移植と改良	融合型知識プログラミング技術
1.高次推論	並列定理証明システムなど
2.知識表現言語	制約論理型言語 CLP 演繹オブジェクト指向DB言語 Quixote
3.実験的応用システム	遺伝子情報処理、法的推論など

中身は、基本ソフトウェアということで、KL1の処理系およびOS、データベース管理ソフト等、知識プログラミングの方は、定理証明や制約言語、オブジェクト指向言語、それから応用などです。

最近、市販のハード、特にスーパーサーバ等がマルチプロセッサ化し、その上のUNIXがプロセス間コミュニケーションをサポートするような新しい機能が追加されつつあります。こういうものがないと移植が大変なわけですが、おかげさまでそういう動向ですから、これなら結構、皆さんに使ってもらえるものが出来そうだと思います。実際、かなりいい性能が出ていて、それについては午後の近山部長の講演等で紹介できると思います。

そういうハードがあれば、ICOTの並列推論マシンなくしても、第五のソフトウェア技術が基盤ツールとして使える、また、知識処理のいろいろなソフトウェアが新たな研究の発火点として使ってもらえるということが期待できる状況になってきました。さらに、データベースなども、単なる並列ではなくてネットワークで結合された分散データベースを扱えますから、データベースのソフトもクライアントサーバのUNIX環境下で動くような仕様を導入して、タンポポの綿毛でいえば、技術移転するときに受け取ってもらいやすいようにするという努力を続けようとしています。

知識プログラミング技術については、知識をこういう言語で書いて下さい、そうするとこういうふうにエンジンが動いて、こういうふうに動きますよという形でまとまるには、まだ2年以上かかるわけです。しかし、この辺はどっちみちやらなくてはならない技術だということで、まずは2年間でさらにまとまりよく仕上げようと思っています。さらに、UNIXの

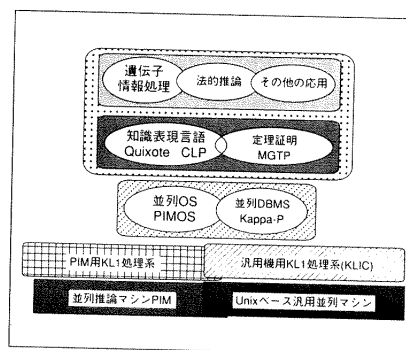
上にあるKLICのような処理系と相性をよくしておいて、継承してくれる人にどんどん受け取ってもらおう。継承してくれる人が計算機の専門家でないこともあるわけですから、そういう人のところに行っても何とか使ってもらえて、そこで新たな芽を出してくれることを期待したいと考えています。

第五世代技術はいつもこのような1枚の絵(図18)にまとめられてきましたが、いろいろ書いてあるテーマの数がだいぶ減りました。このテーマを書いたグループだけが、今ICOTに残っているといいいかもしれません。

まず、一番下は、従来のPIM用のKL1処理系これは、汎用機用のKL1処理系KLICに合うようにどんどん拡張したり、改良したりしていく考えです。

マシンについては、PIMはありますが、今後は世の中のマーケットに出てくるUNIXベースのマシンを対象に考える。ソフトの研究は、現在ある並列推論マシンやPIM用のKL1処理系をどんどん使って、さらに発展させる。発展させつつ、汎用マシンとの相性もいように作り変えていく作業をするということで進めています。

図18 後継プロジェクトにおける研究テーマの構成



最後のスライドを説明します(図19)。

図 19 ICOT 外部との協力体制

第5世代コンピュータ技術の継承とIFSの普及を目指す。 継承先の確保と人材育成が重点課題
1.メーカーの研究グループとの連携 PIMの評価、並列応用、高次推論、知識表現などの研究
2.タスクグループ 大学、研究機関、メーカーの研究者の議論と作業
3.海外共同研究、国際ワークショップ IFSの発展、普及に重点 技術の継承先としても期待 米国 オレゴン大学を追加予定 豪州 オーストラリア国立大学 欧州 SICS（スウェーデン）、ブリストル大学を追加予定

ICOT 外部との協力体制

ICOTは、従来から協力メーカー8社に再委託ということでいろいろな仕事をお願いして、PIMのハードを含め、様々なソフトをつくっていただきました。それをまとめる際に、大学の先生や諸研究所の皆さんにいろいろな形で援助をいただき、一緒に仕事をしてきたわけです。物をつくり続けてきた11年の後、今度はできた物を技術移転するという意味での2年があるわけで、今度の2年のほうが外との関係がより重要になります。ICOTの研究員は減っても、外とのおつき合いはレベルを落とさないように一生懸命努力したいと思っています。

マクロにはどういうことにポイントを置くかということ、とまかく技術を継承してもらわなくてはいけないわけで、そのためにはICOTフリーソフトウェアというような無償公開ソフトウェアをさらに強化していくということが重要です。しかし、それぞれの研究分野で頑張っておられる方にそのままの形で受け取って下さいというのは、なかなか難しいところもあります。そういう場合には、この2年

間で、継承先における人材育成などもやりたい、やらなくてはいけないと思っています。実際、今いろいろと考えて体制を準備しています。従来、メーカーの皆さんには再委託という形で、研究費をこちらから準備して研究していただいていたのですが、後継プロジェクトではそれがなくなってしまうので、今度はメーカーの研究グループは、自前の予算で研究をすることになってしまうわけです。しかしながら、並列処理や知識処理の技術は、将来の重要な技術だと認識いただいているメーカーのグループが多々あります。そういうグループと継続的にPIMの評価や応用、高次推論、知識表現などの研究を続けていく体制を現在作っています。

従来、大学の先生方には非常にお世話になっていますが、ICOTも世代がわりして、ICOTの卒業生、もしくはメーカーで第五世代を担当していた方々の中で三十数人が、主要大学の助教授、教授、また講師になっています。このような人たちはICOTの直系のOBであり、それぞれの大学で、第五世代の研究成果を学生その他に教えるということになっています。この人たちを、従来のICOTの研究員に近い形で動員したいと考えています。まさか毎日来てもらわなくてもいいかなので、ネットワーク経由でお互いにツール、その他をシェアするというので、そのあたりを組織化したいと思っています。そこで、従来の「ワーキンググループ」という名前を、よりインテンシブな仕事をするという意味から、「タスクグループ」という名前に変えて、現在12か13のグループを立ち上げ中です。

海外との関係も、従来はどちらかというと国際親善的なニュアンスが強く、お互いに何を研究しているかを報告し合うというようなレベルにとどまっていたのですが、今後は、一

レベルにとどまっていますが、今後は、一歩進めて、できるだけICOTのフリーソフトウェアを使ってくださいとか、さらに評価をしてくださいというようなことを実際に頼むという形での国際共同研究に移行しようとしています。そのような観点から見ると、やはり相手先もアメリカのナショナルラボ等よりも大学のほうが良いということで、最近、アメリカのオレゴン大学や、並列定理証明の研究等で成果を上げているオーストラリアのオーストラリア国立大学と共同研究を進めています。欧州も、スウェーデンのSICSのほかにも英国のブリストル大学等を追加すべく、現在、準備中です。

このように、第五世代プロジェクトの11年の成果を、後継プロジェクトにおいていろいろな形で継承、発展させようということで、以上のような計画を立て、実行に移っています。ICOTの研究所も、卒業生を送り出して、それ

ぞれの会社に戻った研究者に新たな研究および仕事を続けてもらおうと同時に、新人も入りました。このように世代が交代して、また、今申し上げたような新たな目標に向かって邁進したいと考えているわけです。ICOTの中で何をやっているかというような情報の伝達も、従来は、ともすればマスコミ頼りだったわけですが、現在は電子メール等が非常に細かく張りめぐらされているので、それを利用したいと思っています。今後は、できるだけ電子メールによるダイレクトメールやニュース等の形で、ICOTの活動やソフトの紹介、また、今度こういうKL1のUNIX上の処理系ができたから説明会を開きますというようなニュースを流したいと思っています。

このようなニュースがお目にとまりましたら、ぜひICOTの活動に参加し、またいろいろな形でご協力いただきたいと思います。

ご清聴ありがとうございました。