

第五世代コンピュータ・プロジェクト 最終評価報告書

平成 5 年 3 月 30 日

電子計算機基礎技術開発推進委員会
学術的・技術的評価ワーキング・グループ

目次

はじめに.....	4
学術的・技術的評価.....	5
第五世代コンピュータの概観.....	5
第五世代コンピュータ・プロトタイプ・システムの概要.....	5
注目すべき成果.....	6
並列記号処理技術.....	7
主要技術.....	7
従来技術に比しての優位点.....	8
並列記号処理技術についての当初目標の達成度.....	10
第五世代の並列記号処理技術の意義.....	10
知識情報処理技術.....	11
主要技術.....	11
従来技術に比しての優位点.....	13
知識情報処理技術についての当初目標の達成度.....	14
第五世代の知識情報処理技術の意義.....	14
実験的応用システム.....	15
実験的応用システム構築の目的.....	15
主要なシステム.....	15
主要な評価結果.....	15
総合評価.....	18
プロジェクトが与える影響.....	18
技術的影響.....	18
社会的影響.....	19
将来展望.....	20
汎用的な大規模並列処理技術.....	21
知識情報処理技術の基盤技術.....	21
まとめ.....	23
(資料)第五世代コンピュータ・プロトタイプ・システムの概要.....	24
並列推論マシン PIM.....	25
並列論理型言語 KL1 処理系.....	27
並列推論マシン・オペレーティング・システム PIMOS.....	28
並列非正規関係データベース・システム Kappa-P.....	30
並列定理証明システム MGTP.....	31
並列制約処理言語システム GDCC.....	32

知識表現言語システム Quixote	34
法的推論システム HELIC-II	35
遺伝子情報処理システム.....	37
大規模集積回路設計支援システム	38
自然言語処理システム	40

はじめに

第五世代コンピュータ・プロジェクトは、国際貢献を果たしつつ技術先進国として発展するという我が国の政策のもとに、昭和 57 年度にスタートした。

このような背景から、プロジェクト開始にあたっては、世界を視野にいたれた国家プロジェクトのあり方の議論を行い、国際的にみても創造的・先駆的な技術という意味を込めて「第五世代コンピュータ」と名付けた。そして、その技術目標を「知識情報処理を指向した新しいコンピュータ技術の研究開発」と定め、技術目標に含まれる多くの要素技術の実証・評価を行う必要性から、並列推論型コンピュータのプロトタイプ・システムの試作にその中心を置き、推進母体として財団法人新世代コンピュータ技術開発機構(ICOT)を設立した。昭和 57 年度に研究開発を開始した本プロジェクトも、11 年間で約 540 億を投じ、平成 4 年度をもって終了することとなった。

プロジェクト終了にあたり、平成 3 年 11 月から本電子計算機基礎技術開発推進委員会において、本プロジェクトの評価及び今後の課題と展開のあり方についての検討を進め平成 4 年 6 月には中間報告が出された。

この中間報告においては、まず、「社会的・政策的評価」として、「本プロジェクトは国際貢献に役立つことを明確化したプロジェクトであり、本分野における世界の研究交流の拠点となると共に、国際共同研究を進展させ基礎研究を推進し成果を広く還元したプロジェクトとして、国際貢献を実現した」との結論を下している。更に、「今後の課題と展開のあり方」としては、「本プロジェクトで開発したソフトウェアを研究環境で広く利用できるように既存のコンピュータ上への移植を行うことによって、本分野の研究基盤を構築すると共に、知識処理の基礎となる技術についての更なる研究開発を行うべきである」との提言が示された。本提言を受け、平成 5 年度から 2 年間の予定で、第五世代コンピュータの研究基盤化プロジェクトがスタートすることとなった。

しかしながら、学術的・技術的観点に関しては、平成 4 年 6 月以降も精力的に行ってきたプロトタイプ・システムの研究開発により、中間報告で示した研究内容よりも大幅な進展がみられる。したがって、プロジェクト終了にあたる現時点において最終的な評価を行う必要があり、以下に示す学術的・技術的評価をもって、第五世代コンピュータ・プロジェクトの最終評価とする。

学術的・技術的評価

第五世代コンピュータの概観

第五世代コンピュータ・プロジェクトは、大規模並列記号処理の諸技術、知識情報処理の諸基礎技術の研究開発を行ない、大規模並列知識情報処理のための一貫した技術体系を確立、これに基づく第五世代コンピュータ・プロトタイプ・システムを構築した。また、その上に実用レベルの複雑さと規模を持つ実験的応用ソフトウェアを構築し、技術体系の機能の適切性と実装の効率性を実践的に検証した(詳細は資料 2 を参照)。

第五世代コンピュータ・プロトタイプ・システムの概要

(資料 2:第五世代コンピュータ・プロトタイプ・システムの概要を参照)

第五世代コンピュータ・プロトタイプ・システムは、並列論理型パラダイムに基づく、ハードウェアから機能実証のための応用ソフトウェアにいたるまで、一貫性のあるコンピュータシステムになっている。その概要を以下の表に示す。

ハードウェア	PIM	KL1 の実行に適したアーキテクチャ 総計 1,000 プロセサ以上の 5 モデルを試作・評価 大規模モデルでは 1 秒間 1 億回以上の推論性能
言語処理系	KL1 言語 分散処理系	KL1 言語の並列/分散実行 分散データ管理、分散実行管理に多数の新技术 全モデル共通の言語仕様、各モデルごとの最適化
基本ソフト	PIMOS	PIM の各モデルに共通の OS 徹底した分散管理により管理ボトルネック解消 並列ソフトウェア向きの快適な開発環境
並列記号処理の 基本システム群	Kappa-P	並列データベース・システム 非正規関係モデルによる多様なデータの扱い 分散並列処理による高速処理
知識プログラミング・ソフト	MGTP	並列定理証明システム データベース的応用指向と数学指向の 2 システム 並列処理により世界最高速を達成
並列知識情報処理の 基本システム群	GDCC	並列制約処理言語システム 世界初の非線形代数方程式の取り扱い 容易なプログラム記述と、並列処理による高性能
	Quixote	演繹オブジェクト指向の知識表現言語システム 仮説に基づく推論、不足知識の推論などの高機能 データベースと並列処理による大容量処理能力

機能実証ソフト	HELIC-II	与えられた事件に対して法律適用の可能性を提示 判例データベースの類似検索と法律に基づく推論

技術を実証する 応用ソフト群	遺伝子 情報処理	多数のタンパク質配列間の類似性を解析 独自開発の並列反復改善法による高い性能
	LSI-CAD	論理シミュレーション、セル配置、自動配線 並列処理でスーパーコンピュータ並の性能を実現
	自然言語処理	文脈レベルの文法構造を整備 日本語処理研究開発のための汎用ツール群を提供
	MENDELS ZONE	宣言的記述から KL1 プログラムを自動合成 手書きと同等以上の品質のコードを高速に生成
	碁世代	囲碁の対局を行なう棋士システム

注目すべき成果

プロジェクトは数々の重要な成果をあげたが、以下のものは特に注目すべきものといえる。

世界最高の推論速度

並列推論マシン PIM は、150MLIPS 以上の推論速度を達成した。LIPS(LogicalInferencePerSecond)は1秒間に何回の推論処理が行なえるかを表す単位で、150MLIPS すなわち1秒間に1億5千万回を越える推論速度は汎用大型コンピュータの約100倍にあたり、現在世界最高のものである。

応用ソフトウェアに対する高い実効並列処理効果

単純なプログラムのピーク時の並列処理効果だけでなく、実用的な複雑さを持つ応用プログラムに対しても高い実効並列処理効果を得た。たとえば並列定理証明システム MGTP では256プロセッサ・システム上で200倍以上の並列処理による速度向上を示し、アルゴリズム上の工夫とあいまって、従来高速とされてきたワークステーション上の自動証明器の400～1000倍の性能を達成した。

並列ソフトウェアの高い生産性

第五世代技術を用いることによって、実用レベルの規模と複雑さを持つ問題に対する並列処理ソフトウェア構築が、従来の手法では考えられなかったほどの短期間で可能になった。たとえば、従来実用的なオペレーティング・システムの開発には最低でも数年間を要していたが、本格的な並列処理システム用のオペレーティング・システムである PIMOS の最初の版(KL1 で約44,000行)は、実験機 Multi-PSI の完成から約半年で開発を完了している。さらにこの PIMOS が提供する諸機能を利用した応用システムの開発、たとえば並列 LSI-CAD 実験システムの一部である論理シミュレータの開発では、パーティクルタイム法という並列処理向きの複雑なアルゴリズムを用いているにも関わらず、約3

人月というわずかな工数で開発できた。これは、従来技術による並列ソフトウェア開発の場合の10分の1以下であり、第五世代技術のもたらす高い生産性を実証したものである。

未解決の定理を証明

PIM上に構築された並列自動定理証明システムMGTPは、人類にとって未解決だった群論に関する問題の一部を自動証明することに成功した。かつてコンピュータによる数値処理は天文学者を大量の計算から解放し、より創造的な科学的思索への集中を可能にし、天文学に多大な進歩をもたらした。MGTPの成果は、第五世代技術が数学者を繁雑な証明手続きから解放し、より創造的な思索に専念させ、数学を飛躍的に進歩させる原動力となる可能性を示している。

論理に基づく技術による知識情報処理の実用性を実証

数理論理という普遍的な原理に基礎をおいた知識情報処理技術が、並列推論技術と結びつくことによって、実用的な規模と複雑さを持つ問題に適用して、十分な機能と性能を実現できることを実証した。たとえば、論理に基づく知識情報処理技術を利用して組み上げた機能実証ソフトウェアのひとつである法的推論システムHELIC-IIは、法律と過去の判例の両者を用いて新しい事件に対するさまざまな法律適用の可能性を提示する、これまでにない機能を持つシステムであるが、並列推論技術により実用的な速度を実現している。

このように、第五世代技術は次世紀のコンピュータ技術全般の基礎となりうる並列推論技術を、実証的に確立したといえる。

並列記号処理技術

主要技術

第五世代プロジェクトで開発された並列記号処理の主要技術には、以下のものがある。

並列論理型パラダイムによる一貫した技術体系

並列記号処理技術のすべての階層の共通基盤として並列論理型パラダイムを設定し、並列処理を前提とした一貫した技術体系を構築した。

並列記号処理言語設計・実装技術

並列記号処理ソフトウェアの構築を難しくしている要因を同定し、それを取り除くために必要なプログラム言語機能を備えた、簡潔で高機能なプログラム言語KL1を設計した。また、KL1の効率的実装に必要な諸技術を確立し、実用的な高性能の並列記号処理

システムを設計・実装、ソフトウェア研究開発の共通ツールとして提供した。

並列記号処理ハードウェア技術

並列ソフトウェアの研究開発に向けた世界最大級規模の並列処理ハードウェアを設計・実装し、高性能並列記号処理言語システムのためのハードウェア技術を確立した。

並列記号処理ソフトウェア開発環境技術

世界初の実用的な大規模並列記号処理ソフトウェア開発環境であるオペレーティング・システム PIMOS を設計・実装し、ソフトウェア研究開発に提供・評価した。

並列記号処理ソフトウェアのプログラミング技術

並列論理型言語による大規模並列システム向きのプログラミング技法・ソフトウェア構成技法を開発、オペレーティング・システムや応用プログラムの構築において実践した。

従来技術に比しての優位点

第五世代プロジェクトで開発された並列記号処理技術は、以下の各点で従来技術に優っているといえる。

並列処理を前提としたポリシーの一貫した記号処理システム

逐次処理技術の漸時改良による従来技術は複雑で統一性に欠け、並列ソフトウェアの開発は困難を極めた。これに対し第五世代技術は、並列処理を前提とした並列論理型パラダイムに基づく一貫した技術体系であり、大規模並列ソフトウェアの開発を飛躍的に容易にした。

並列記号処理言語設計・実装技術

逐次処理言語に必要な応じて並列機能を追加した従来の並列プログラム言語に比べ、並列処理を前提とした設計による並列論理型言語 KL1 は、大幅な機能向上と簡潔な仕様を同時に実現した。また、知識情報処理に不可欠な非定型的処理や、負荷分散の容易な細粒度並列処理の効率を大きく向上させる実装技術を開発し、ソフトウェアの負担を大幅に軽減させた。

並列記号処理ハードウェア技術

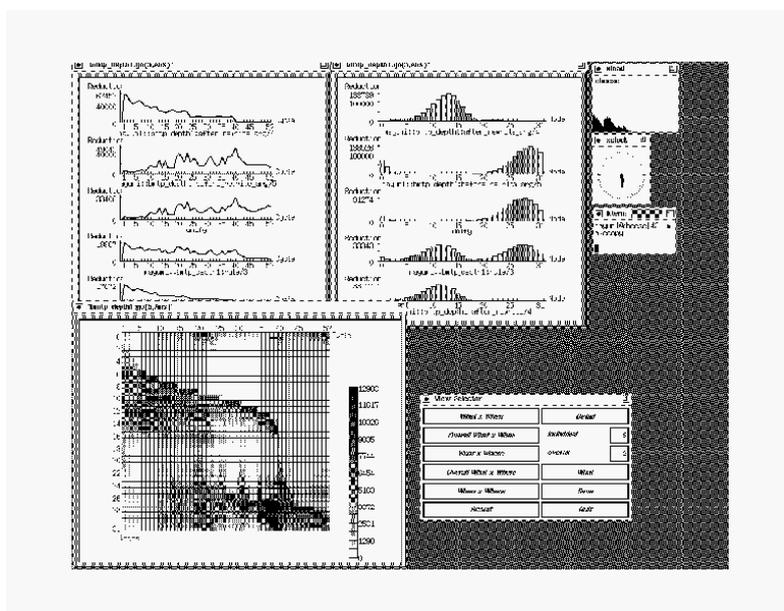
従来の定型的な並列数値処理に向きの技術とは異なる、並列論理型言語の細粒度並列処理向きのハードウェア技術を確立した。

たとえば、512 台のプロセッサを持つ PIM モデル p ではピーク性能で 156MLIPS(1 秒

間に 1 億 5600 万回の推論処理を行なう速度)、256 台のプロセッサを持つモデル m では 153MLIPS という、汎用大型機の約 100 倍に当たる世界最高の推論処理速度を達成している。実際の応用システムでの利用でも、アルゴリズムの工夫とあいまって、512 台にいたるまでほぼプロセッサの台数に比例する並列処理効果を得ている。

大規模並列記号処理向きソフトウェア環境

従来技術での並列処理ソフトウェア環境は、小規模な並列処理や定型的な並列数値処理のためのソフトウェアの開発のみを対象としたものであった。これに対し、第五世代プロジェクトで開発したオペレーティング・システム PIMOS は、徹底した分散管理と並列処理向きの開発ツール群の提供によって、大規模並列ソフトウェア研究開発を快適に行なえるプログラミング環境を実現した。



PIMOS のプロファイラによる表示例

たとえば、PIMOS が提供するデバッガは、KL1 言語処理系の機能を利用した自動的なデッドロックの検出や、任意のプログラム部分の実行の一時停止・再開、別ウインドウを用いてのプロセス別のトレース、次第に生成されて行くデータの監視などができる。同じく PIMOS が提供するプロファイラは、プログラム実行中あるいは実行後に、どのような計算がいつ、どこで(どのプロセッサで)行なわれているのかをグラフィックに表示し、負荷分散状況の容易な把握を可能にしている。

並列記号処理プログラミング技術

従来のプログラミング技術は、逐次処理を前提とした記号処理技術や、処理の単純性・定型性に依存した並列処理技術であった。これに対し第五世代技術は、並列記号処理言

語の高い機能、その効率的な大規模並列実装、その上のオペレーティング・システムが提供する快適な開発環境を利用し、複雑で非定型な問題を効率的に並列処理できる。

たとえば、プロジェクト内で開発された互いに通信し合うプロセスを並列論理型言語で表現する技法は非常に汎用性が高く、PIMOS を始めとするほとんどの KL1 言語によるソフトウェア・システムで用いている。この技法をプログラム言語として再構成した KL1 の上位言語 AYA も開発された。また、特定の問題ではなくプロセス構造に着目した汎用性の高いさまざまな負荷分散技法を設計、負荷分散ライブラリとして応用システムの開発者に提供した。このライブラリを用いた負荷分散は、多くの応用システムの並列化の基礎となった。

並列記号処理技術についての当初目標の達成度

並列記号処理技術については、プロジェクトの当初目標ではソフトウェアとしての問題解決・推論メカニズムとハードウェアとしての問題解決・推論マシンが開発テーマとして挙げられていた。

問題解決推論メカニズムについては、KL1、PIMOS 等により、当初目標が十分に達成されている。また、問題解決・推論マシンについては、PIM により、当初目標の 100MLIPS(LIPS:LogicalInferencePerSecond、1 秒間に何回の推論処理が行なえるかの単位)を大きく上回る、150MLIPS 以上の処理能力を達成している。

第五世代の並列記号処理技術の意義

従来の並列コンピュータシステムの研究開発は、定型的な数値処理が必要な分野を主たる問題領域として設定して進められてきた。その結果近年では、この分野については逐次型を基本とする処理方式に取って代わらんとする勢いを示している。しかし、非定型的な処理を必要とする記号処理についての並列処理の研究開発は、第五世代プロジェクト以外では多くの努力を払われてこなかった。このため、記号処理が必要な応用分野の研究開発については、逐次処理方式を取り続けてきたため、計算処理速度の壁にぶつかって十分な進捗が得られない局面も多くなってきた。

第五世代の技術は世界に先駆けて並列記号処理技術体系を確立し、実験的な応用システムを通して、その有効性を実証的に示した。この並列記号処理技術は、将来発展が見込まれる本格的な大規模知識情報処理に必要な、並列処理の容易な記述と十分な性能とを提供する、欠くべからざる基礎技術となるものである。これはまた、数値処理等の他分野に応用しても、新たな技術展開の基礎となり得る技術である。

知識情報処理技術

主要技術

第五世代プロジェクトで開発された知識情報処理の主要技術には、以下のものがある。

並列論理型パラダイムに基づく知識情報処理基礎技術

知識情報の表現・管理・利用の諸技術を、並列論理型パラダイムに基づいて再構成し、高いレベルで自由に表現した知識を効率的に並列処理する、大規模知識情報処理に必須な基礎技術を確立した。

知識表現技術

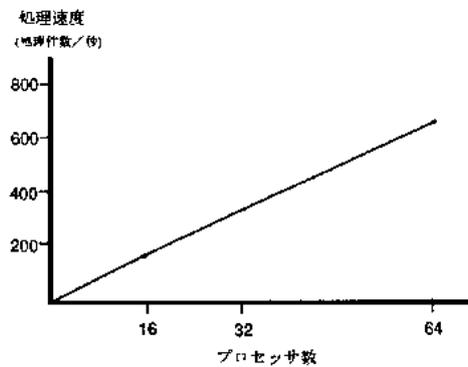
高レベルで多角的な知識表現技法・知識表現言語を開発し、知識表現の自由度を大幅に増した。また、これを具体的問題に適用することによって、人間の知識のさまざまな性質を、計算機処理の側面から明らかにした。

たとえば、制約論理型言語 **GDCC** は、どのような問題を解くのかを記述すれば、解法をプログラミングせずとも、回答を得ることができるような言語になっている。場面に応じた問題解決をシステムが行うために要求される多大な計算処理は、並列処理を導入することによって効率化され、その上での種々の応用システムの試作を通じて高い機能と高い効率の実現が確かめられている。

また、知識表現言語 **Quixote** は、通常の演繹の他に、実世界の知識を自然に表現するために必要な知識ベースに仮説を追加して推論する機能、知識ベースに不足している知識を推論する機能なども含めた、包括的な機能を持っている。その有効性は、新しい事件の判決を予測する法的推論、複数の自然言語表現から曖昧性を除去する状況推論、複雑な構造体間の関連を推論する遺伝子情報処理などの応用によって確かめられている。

知識情報管理技術

高レベルで多角的に表現した大量データの効率的管理の基礎技術を確立し、大規模並列知識ベースシステムのモデルとして提示・試験実装し、機能の適切性・実装の効率性を確認した。

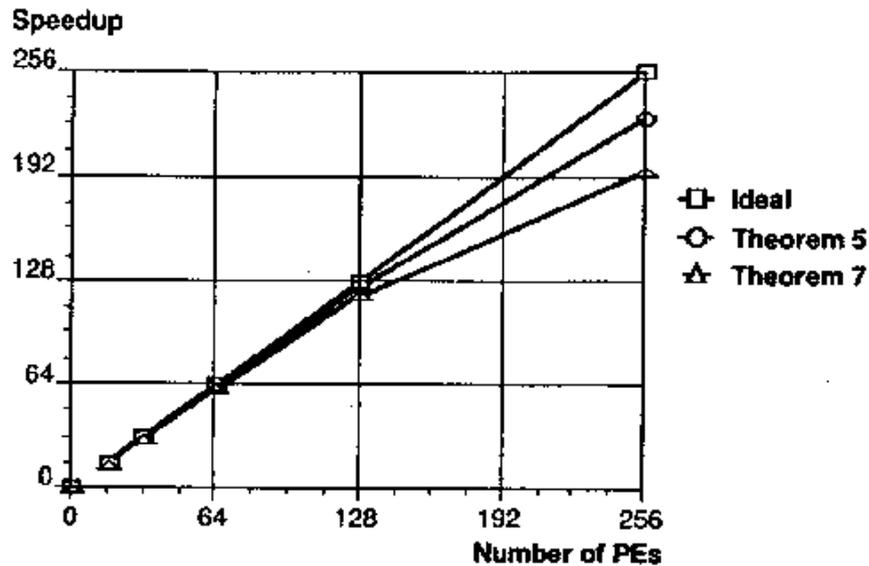


モチーフ検索の並列処理効果

7000 たとえば、並列データベース・システム Kappa-P による分子生物学データベースからのモチーフ検索では、用いたプロセッサ数にほぼ比例する性能向上を得ている。この検索は、従来の関係データベースでも処理可能な単純な検索とは異なり、生物学上の知見に基づいた複雑な類似検索である。Kappa-P はこのような高度な検索も、簡単に指定できるシステムになっている。

並列知識利用基礎技術

論理式、制約などさまざまな形式で表現した情報の、効率的並列処理技術の基礎を確立、これを並列処理システム上に実装した。また、通常の推論の枠にとられない高次推論技術、学習技術の理論的基礎も確立した。



MGTP の並列処理効果の例

7000 たとえば、並列自動定理証明システム MGTP においては、200 倍以上(256 プロセッサ使用時)の並列処理による性能向上を達成している。これは、従来高性能であるといわれてきたワークステーション上のシステムに比べても 400~1000 倍の性能であり、人類にとって未解決だった群論に関する問題の一部を、自動的に証明することにも成功している。

従来技術に比しての優位点

第五世代プロジェクトで開発された知識情報処理技術は、以下の各点で従来技術に優っているといえる。

高レベルで表現した知識を効率的に処理する基礎技術

第五世代技術は、従来技術では不可能だった高レベル表現の情報を効率的に並列処理する基礎技術を確立し、大規模で実際的な問題に対する知識情報処理への道を開いた。

高レベルで自由度の高い知識表現技術

第五世代技術は、従来技術では不可能だった高レベルで自由な知識表現の基礎技術を確立、実際に各種の問題についての適用を通じて、人間の持つ知識のさまざまな性質を、計算機処理という側面から明らかにした。一方、高レベルの表現は、計算機による知識

獲得などのさらに高いレベルの処理も著しく容易にした。

大容量知識情報の効率的な並列管理技術

第五世代技術は、従来の技術では効率的に扱えなかった非定型的な大量データを効率的に管理する並列分散データベース技術を確立し、実用的な問題への計算機による知識情報処理の適用に不可欠となる、自由に表現した大量の情報の効率的アクセスへの道を開いた。

高い自由度で表現した知識の並列処理基礎技術

第五世代技術は、従来の技術では困難だったさまざまな形式で自由に表現したデータを効率的に並列処理して利用するための基礎技術を確立し、大規模な知識情報の効率的利用の基礎を作った。

知識情報処理技術についての当初目標の達成度

知識情報処理技術については、プロジェクトの当初目標では、知識ベースのソフトウェアとしての知識ベースメカニズムと、ハードウェアとしての知識ベースマシンがテーマとして挙げられていた。また、知的インタフェース・システムがテーマとして挙げられ、自然言語処理の他、音声・画像処理も含まれていた。これらのインタフェース・システムのテーマは、中核技術の上に築かれ、その検証にも役立つテーマと位置付けられていた。

知識ベースメカニズムについては、Kappa-P によって実現されている。また、知識ベース・マシンについては、PIM がファイルシステムを備えることで、当初目標が達成されている。

知的インタフェース・システムについては、音声や画像を数値的なデータとしてではなく知識情報として扱うためには、膨大な背景的知識の蓄積と構造化が必要であることが、研究の進捗と共に明らかになった。このため、プロジェクトの中間段階からは自然言語処理を中心に据え、背景的知識の蓄積と構造化のための基礎技術、すなわち、論理を共通基盤とする知識表現と、高い自由度で表現した知識の並列処理に的を絞った研究開発が行なわれてきた。具体的なインタフェース・システムの開発にまでは至らなかったが、基礎技術の研究開発成果は実験的応用システムに適用され、知的インタフェース技術の基盤技術として有効であることが実証された。

第五世代の知識情報処理技術の意義

プロジェクト発足当時、知識情報処理技術の急速な発展に寄せる期待は大きく、比較的短期間に本格的な知識情報処理システムが実用化できるだろうという見通しに基づいて、機械翻訳システムなどに対して民間でも多大な開発努力が払われた。これらの研究開発の動向を見ると、現段階の基礎技術に基づいたシステムでは適用範囲の限定が必須で、汎用

的なシステムの実現には知識処理の基盤技術の一層の成熟が必要であることがわかってきている。

第五世代プロジェクトでは、非常に汎用的な体系である数理論理に基づく知識表現、管理、利用技術を、並列処理の提供する高い処理能力と組み合わせることによって、実用的な知識情報処理システムの共通基盤となし得ることを示した。これによって、自然科学、社会科学、人文／認知科学等の異なる分野における知識を、論理を基盤とする知識表現言語を用いて表現し、これらの諸分野で必要となる知識の質の比較を可能にし、将来の知識情報処理技術の研究開発の方向の策定に不可欠な座標系を示すことができた。

この成果は知識情報処理の基盤技術を示した点において大きな意義を持つが、本格的な実用システムの構築に至るまでには解決すべき問題がまだまだ数多く残されている。今後、本プロジェクトが確立した基盤技術に基づいて、一層の研究開発を進めることが必要である。

実験的応用システム

実験的応用システム構築の目的

第五世代技術のような新たな基礎的技術の体系の有効性を示すには、技術体系自身についての検討評価のみでは不十分で、実際にその技術体系を用いて応用システムを構築することを通して検証しなければならない。実験的応用システムの構築の本来の目的は、応用システムの構築自体ではなく、それを通じて第五世代技術体系を評価することにあつた。

主要なシステム

構築した実験的応用システムの主なものとしては、以下のものがある。

法的推論:	法令の論理的解釈、知的判例検索 要素技術:並列記号処理、知識表現・管理、自動定理証明、・・・
設計問題:	電子回路の論理シミュレーション、回路自動配置・配線、 ロボット設計支援システム 要素技術:並列記号処理、知識の制約表現、並列制約処理、・・・
遺伝子情報処理:	アミノ酸配列解析、モチーフ抽出、立体構造予測 要素技術:並列記号処理、負荷分散
自然言語処理:	汎用日本語処理系、談話処理、並列自然言語処理 要素技術:意味表現、知識表現・管理・利用、並列記号処理、・・・
プログラム合成:	仕様記述からの並列プログラム自動合成 要素技術:等式論理、時相論理、ペトリネット、・・・

主要な評価結果

第五世代技術を具体的な問題に適用し、その総合的な優位性を実践的に検証

プロジェクトで開発した並列記号処理技術、知識情報処理技術を実用レベルの規模と

複雑さを持つ具体的な問題に適用し、これらの技術を総合的に活用することによって、これらの技術が提供する機能の適切性、実現手法の効率性を検証した。また、従来技術でも同様の機能を実現している分野については、従来の手法での実現方法と比較し、第五世代技術の優位性を実践的に検証した。

並列記号処理技術による高い実行性能と開発効率

実験的応用システムは並列推論システム上で動作するよう開発した。並列推論マシンの性能を生かすことによって、逐次処理方式では達成し難い高い実行性能を得ることができた。また、並列記号処理を前提として設計されたプログラム言語と、並列処理のために設計されたソフトウェア開発環境を利用できたため、非常に高いソフトウェア開発効率を得ることができた。

従来も同程度の困難さを伴う問題を扱う技術はあったが、処理効率やソフトウェア開発環境の問題から、机上の検討や小規模の実験のみに終ることが多かった。これに比べ、第五世代コンピュータ・プロジェクトにおいては、プロジェクトで構築した高性能の並列推論マシンと快適なソフトウェア開発環境を利用できたため、実用レベルの複雑性と規模を持つ問題を解決することができた。

たとえば並列 LSI-CAD 実験システムの一部である論理シミュレータでは、問題が必ずしも並列処理に向いていないにも関わらず、256 台のプロセッサで 166 倍の高い並列処理効果を得ており、絶対速度としてもスーパーコンピュータに匹敵する高い性能を得ている。この並列性能を得るためには、バーチャルタイム法という複雑なアルゴリズムを用い、種々のチューニングを施す必要があったが、並列推論システムの提供する諸機能を利用することによって、約 3 人月という、従来の方式による並列プログラミングに比べてひと桁以上少ない工数での開発が可能だった。また、他の並列アルゴリズムについても同様に少ない工数で開発できたため、実験的な比較検討も可能になった。

知識情報処理技術による高レベルでコンパクトな記述

従来の知識情報処理応用システムの研究開発では、特定の応用に対する技術開発のみが行なわれることが多く、ある特定の問題にしか適用できない技術の開発に重点が置かれがちであった。このような技術は他の分野に応用しにくいだけでなく、選んだ対象領域の周辺においてすら、問題がわずかに異なるだけで適用が困難な場合も多かった。また、応用に関わるすべての問題を解決しようとするため、必要な開発項目が多くなり、高度なシステムの開発には困難を極めた。

これに比べ、第五世代コンピュータ・プロジェクトにおいては、並列論理型という一貫した枠組の中で研究開発を行なってきたため、応用システム実現のための要素システムとして、基礎的な知識情報処理技術の研究開発成果を順次取り込むことができた。これによって、応用システムでは非常に高いレベルのアルゴリズム設計とソフトウェア記

述に集中することができ、従来開発が困難だった高度な機能を持つシステムを、短期間で開発することができた。

たとえば、法的推論システムの扱う法律の分野では、その体系知識(法令文)が完全でないため、過去の運用実績(判例)を補わなければ実問題が解決できない。そのため、法的推論システムの実現には、法令文の論理的記述とそれを用いた演繹的推論、判例の自然な記述と類比による推論という、二種類の知識情報処理技術が必要である。このような複雑な問題解決システムであるにも関わらず、推論モジュールの第1版は約4,500行と、従来技術に比べてひと桁以上コンパクトな記述となり、開発工数も9人月程度で済んでいる。このようなコンパクトな記述と短期間で開発を可能にしたのは、プロジェクト内で開発した要素技術に基づき並列推論マシン PIM 上に実現した、効率的で汎用性の高い要素システムの利用によるものである。法令文の処理には並列自動定理証明器 MGTP を利用して理論的裏付けのある論証モジュールを構築し、後者の判例の処理には、事例ベース推論や類推などの高次推論技術と、自然言語処理にも共通する知識表現技術を応用し、並列処理による効率的な類比検索モジュールを構築している。

総合評価

ここでは、第五世代コンピュータ・プロジェクトについての、これまでの議論をまとめるとともに、その総合的な意義について述べる。

プロジェクトが与える影響

技術的影響

並列記号処理技術(汎用的な大規模並列処理技術)

従来の技術では、知識処理の問題の高度並列処理は一般に困難とされていた。しかし、本プロジェクトで開発された技術は、並列処理に用いるプロセッサの台数を大幅に増加させ、それらのプロセッサを有効に稼働させる得ることを示した。実際に、千台規模までのプロセッサ数について、ほぼその台数に比例した処理速度が得られることを確認し、さらに、数十万台の規模についても有効に稼働させ得る見通しを得た。また、この技術によれば、大規模な並列ソフトウェアを効率良く作成可能であることを、多くの実験的応用システムの作成を通して実証した。

このような汎用的な大規模並列処理の技術は、近く到来する本格的な並列処理、分散処理の時代に向けて、並列ソフトウェア作成に関するボトルネックを解消し、ソフトウェア危機の解決に大きく貢献できることを示した。さらに、この技術によって、従来のノイマン型コンピュータの数百から数千倍の処理能力の実現方式が確立したことから、述語論理に基づく高次推論機構の実現、本格的な知識ベースの作成、知的対話処理の実現など、従来技術では到達不能であった研究領域へと踏み込むことが可能となった。

知識情報処理技術の基盤技術

上で述べた大規模並列処理技術を土台として、知識情報処理の基盤技術、すなわち、論理に基づき各種の知識をプログラムし、知識ベース化するための中核技術が確立した。この技術は、人間社会に存在する各種の有用な知識を統一的に記述し、コンピュータ上で利用するための技術である。

論理に基づく知識の記述は、従来は、人間社会における規則や経験を客観的、形式的に表現し、整理する目的で使われることが多かった。上で述べた大規模並列処理技術によって、述語論理で記述された定理を、従来技術に比べて、数百倍以上高速に証明することが可能となった。これは、同様に、述語論理に基いて記述された知識を一種のプログラムと見なして、コンピュータ上で高速に実行し解答を得ることが、実用的な時間内で可能となったことを意味する。

この技術によって、コンピュータを、論理に基づく知識表現言語を解釈し、ユーザの

問いかげに応答する高度な推論マシンとすることができる。将来的には、ユーザは、マシンのハードウェアやソフトウェアの内部構造を知らなくとも、論理に基づき知識をプログラムするだけで、高度な知識情報処理システムを作成できることとなる。この技術を発展させると、コンピュータの細かな専門的知識の代わりに、この知識表現言語を学ぶことで、医学や法律分野の知識ベースやその応用システムなどの作成が可能となり、プログラミングやソフトウェアといったものの方法論が変わる可能性がある。

その他の技術

第五世代コンピュータ・プロジェクトの開始当初の計画には、その技術目標の実現に関連すると思われるハードウェアやソフトウェア、さらに知識処理の応用など、いろいろな研究課題が挙げられていた。それらの中には、柔軟なマンマシンインターフェース実現を目指した画像や音声処理、自然言語処理技術等も含まれていた。

これらの研究課題は、研究の進捗とともに評価され、相互に比較されて、徐々に絞り込まれた。最終的には上で述べた、中核的な目標との関連において最も重要と思われる二つの大きな技術のグループにまとめられた。

一方、実験的応用システムとしては、当初の関連研究課題に挙げられていなかった遺伝子情報処理システムや、判例をもとに被告の罪状を推論する法的推論システムなどの新しいシステムの試作が追加された。実験的応用システムは、本プロジェクトで開発された並列プログラミング技術や知識表現技術などの評価を効果的に実施することを可能とした。

また、その開発を通して、知識を記述したり体系化する際の難易度が、応用分野ごとにかなり異なることもわかってきた。たとえば、自然科学分野に比べ、社会科学分野や人文科学分野では難易度が高く、さらに深い基礎研究が必要である。そのほか、実験的応用システムは、それ自身が新しい応用分野を開くという重要な役割を演じた。

社会的影響

第五世代コンピュータ・プロジェクトが開始された頃には、日本のコンピュータ産業が発展し、創造的な技術開発や国際貢献が期待された。さらに、その波及効果として世界に通用する研究者の育成や、それによる日本の基礎研究能力の向上なども期待された。

11年間の研究開発成果と、それを産み出す過程における国際的研究活動の展開、およびそれらについての国際的な評価を振り返るとき、本プロジェクトは、このような期待の多くのものに答えることができたといえる。

「知識情報処理に適した新しいコンピュータ技術体系の確立」という未踏の技術目標は、世界の若い研究者の知的興味とロマンをかきたてることに成功した。また、その技術目標は、リスクの大きな創造的技術開発を基礎研究段階から行うものであったことから、国のプロジェクトとして実施するにふさわしいものであった。

その目標を達成するためには、多くの理論研究を必要とすると共に、そのプロトタイプ実現の過程では、工学的な技術蓄積を必要とした。本プロジェクトでは、理論研究と工学的技術開発とのバランスのとれた運営に成功したといえる。

技術目標の魅力の高さと簡明さは、国際協力と貢献を行なう上での重要な条件であり、これを満たしていたことから、国際協力も順調に進展した。国際交流は、前期(昭和 57～59 年度)の研究者の個人レベルの相互訪問から始まり、中期(昭和 60～63 年度)の定期的な 2 国間ワークショップの開催へと続いた。このレベルで培った、草の根的な研究者間の協力関係をもとに、後期(平成元～4 年度)に入ってからには研究用のツールを共用した共同研究へと発展し、この段階で、政府機関を含む組織的な協力関係が確立した。

当初から研究内容を内外無差別に公表してきたことも、国際交流の活発化の重要な要因であった。海外研究者の受け入れによる交流、ソフトウェアの無償公開等、研究内容を目に見える形で公表してきていることは、今後の国が行う研究開発のモデルとなるものと言える。

このような世界に開かれた研究環境のもとで、本プロジェクトに参加した多くの研究者は、新しいコンピュータの理論やソフトウェア、ハードウェア技術を発展させてきた。海外研究者との共同作業は、日常の研究活動の一部となり、文化の違いや研究者の社会的環境の違いを乗り越え、世界に通用する研究者を輩出するに至っている。

このような状況を捉えて、フランスの国立研究所 I N R I A のジル・カーン博士は「第五世代コンピュータ以前は、日本の技術は見えても研究者の顔は見えなかった。しかし、第五世代コンピュータ以後は、ヨーロッパの研究者も、日本の研究者との個人的な親交を持つことで、顔が見える状況となった。」と評している。

以上のように、本プロジェクトは、創造的な技術開発、国際貢献、および、日本の基礎研究能力の向上など、その当初の期待に十分応え、日本のナショナルプロジェクトのモデルを示し得たと考えられる。

将来展望

第五世代コンピュータ・プロジェクトの技術的な成果を、次の 2 つに大別して論じてきた。

a) 並列記号処理技術(汎用的な大規模並列処理技術)

b) 知識情報処理技術の基盤技術

これらの技術は、第五世代コンピュータ・プロジェクトでは、階層をなしており、並列記号処理技術が土台となり、その上に知識情報処理技術が構築されている。並列記号処理技術による処理能力の大幅な向上は、知識情報処理技術の到達可能な上限を押し上げる形となっている。知識情報処理技術のさらに先には、本格的な人工知能の実現へと連る技術がある。

第五世代コンピュータ・プロジェクトで開発された知識情報処理技術は、従来の技術では、到達困難であった知識情報処理の応用分野のいくつかに足を踏み入れることを可能とした。その結果、その先の研究分野について、研究の道筋が見えるものや、さらなる基礎研究が必要なものなどの区別が、明確になりつつある。このようなものの中には、パターン認識や学習などの、人間のより高度な知的活動に関する研究がある。

汎用的な大規模並列処理技術

本プロジェクトで開発された汎用的な大規模並列技術は、並列推論マシン(PIM)、その言語である並列論理型言語(KL1)、そのオペレーティング・システム(PIMOS)が中核となっている。

並列推論マシンは、512台の要素プロセッサを有するモデルを始めとして、総計で1,000台を越える、5種のモデルが試作された。これらのモデルは、共通の言語とオペレーティング・システムで制御され、その上に、遺伝子情報処理システムなど、約20種の実験的応用システムが試作されている。これらのシステムの開発によって、大規模並列処理技術が、記号処理や知識処理の応用に対して、世界で最も高速なシステムを実現しているとともに、並列ソフトウェアの生産性についても、他の追随を許さないものであることが実証された。

このような大規模並列処理技術は、プロジェクトの開始当初では、開発リスクの極めて大きな夢の技術として認識されていた。しかしながら、10余年を経た今日、多くの人々が、並列処理技術に基づくコンピュータが次の時代の主流であると認識するようになり、現実にそのようなコンピュータが、市場に次々と登場する時代となった。

これらのコンピュータは、まだ、プロセッサの台数やメモリー量など、そのハードウェアの規模は十分ではないが、本プロジェクトで開発された言語や基本ソフトウェアを搭載することが可能であり、それにより、知識情報処理の応用に対して実用的な処理速度を達成できる潜在的な能力を有している。これらのコンピュータに、本プロジェクトで開発された言語や基本ソフトウェアなどを移植し、多くの研究者や技術者に広く利用可能とすることが求められている。

現在、このための、「第五世代コンピュータの研究基盤化プロジェクト」が準備されている。

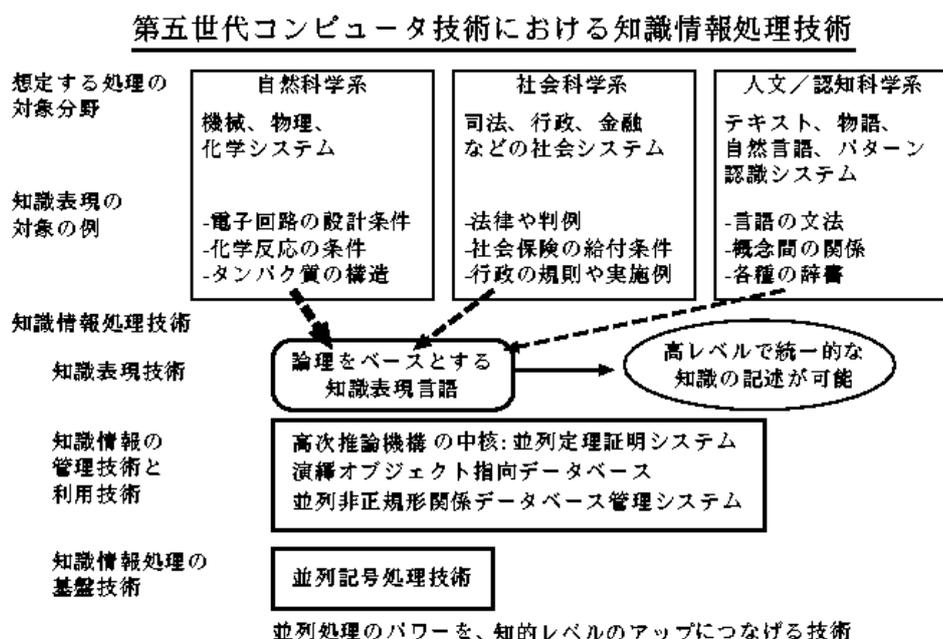
知識情報処理技術の基盤技術

本プロジェクトで開発された知識情報処理の基盤技術、すなわち、論理に基づき各種の知識をプログラムし、知識ベース化するための中核技術は、大規模並列処理技術を土台として構築されている。

その土台となっている大規模並列処理技術が、「第五世代コンピュータの研究基盤化プロジェクト」の実施によって、広く世界に普及していくことは、知識情報処理技術の将来に

大きな可能性を生み出すことになる。

本プロジェクトで開発された論理を基本とした知識表現言語は、演繹・オブジェクト指向言語や制約論理型言語と呼ばれている。この種の言語は、人間社会にとって有用な知識を記述し、知識ベース化する場合の最も有望な枠組みである。



このような知識表現言語や関連技術は新しい応用分野の開拓を加速する。自然科学分野でも、従来の機械や建築などの分野に加えて、分子生物学における遺伝子情報処理など、新しい分野へ応用され始めている。また、社会科学分野における行政や法律の知識情報処理が注目されている。さらに、いくつかの実験的応用システムの試作を通して、論理に基づく知識表現言語が、従来型の知識表現言語と異なり、多くの分野に統一的に適用可能であることがわかってきた。

それと同時に、人間社会にとって有用な知識を活用するためには、知識表現言語で記述し、コンピュータ上に蓄積する作業とともに、その構造や体系自身をより整理された形に組み直す作業が必要であることもわかってきた。

大規模並列処理技術とそれに裏打ちされた知識情報処理技術が、自由に利用可能となる時代には、知識の整理や体系化を含む知識ベース化技術など、新しい知識情報処理技術が生まれ、その分野特有の発展を遂げるであろう。

このような発展段階に至った時に、第五世代コンピュータの知識情報処理技術の中心である「論理に基づく知識表現をプログラミング言語とし、その証明過程を計算のモデルとする新しいコンピュータ技術」が、真のブレークスルーを達成していたことがわかるだろう。

まとめ

本プロジェクトは、当初に定義した『知識ベースを用いる推論を中核メカニズムとする新しいコンピュータ技術』という、革新的なコンピュータ技術を開発し、長期に渡って発展を続ける、新しいコンピュータの世代を切り開いたといえよう。

(資料)第五世代コンピュータ・プロトタイプ・システムの概要

第五世代コンピュータ・プロトタイプ・システムは、主として以下の要素システムからなる。

1. 並列推論マシン
並列論理型言語 KL1 の実行のために最適化したアーキテクチャを持つ、並列計算機システム
2. 並列論理型言語 KL1 処理系
並列論理型言語 KL1 を効率的に並列に実行するための、プログラム言語システム
3. 基本ソフトウェア
並列推論マシン上での KL1 プログラムの実行を司る、もっとも基本的なソフトウェア。以下のふたつのシステムからなる。
 - ① 並列推論マシン・オペレーティング・システム PIMOS
並列推論マシンのハードウェアを効率的に管理し、並列プログラムの開発に適したソフトウェア開発環境を提供するオペレーティング・システム
 - ② 並列非正規データベース・システム Kappa-P
並列推論マシン上に知識情報処理に必要なデータベース機能を提供するシステム
4. 知識プログラミング・ソフトウェア
知識処理応用ソフトウェアで共通に利用できる基礎的な知識情報処理技術を提供するシステム群。代表的なものに以下の 3 システムがある。
 - ① 並列定理証明システム MGTP
モデル生成型の並列自動定理証明システム
 - ② 並列制約処理言語システム GDCC
制約論理型言語の並列処理システム
 - ③ 知識表現言語システム Quixote
大量で複雑な知識を計算機上で効率良く扱うための知識表現言語システム
5. 機能実証ソフトウェア
並列推論マシンに構築した応用ソフトウェア・システム群。KL1 言語処理系や知識プログラミング・ソフトウェアの諸機能を利用して構築されており、これらのシステムの機能を実証するものである。代表的なものとしては以下の 5 システムが挙げられる。

- ① 法的推論システム HELIC-II
新しい事件を対して、判例を用いてさまざま法律適用の可能性を提示するシステム
- ② 遺伝子情報処理システム
タンパク質配列の類似性解析行なうシステム
- ③ 大規模集積回路設計支援システム
LSI-CAD の工程中、論理シミュレーション、LSI セル配置、および LSI 配線の工程を並列に実行するシステム
- ④ 自然言語処理システム
日本語処理システムの研究開発のベースとなる汎用日本語処理系のツール群を提供するシステム
- ⑤ プログラム自動合成システム MENDELZONE
宣言的記述から KL1 プログラムを自動合成するシステム
- ⑥ 棋士システム碁世代
囲碁の対局を行なう棋士システム

以下、それぞれについてやや詳しく述べる。

並列推論マシン PIM

並列推論マシン PIM は、FGCS プロトタイプ・システムの基本部分を構成する。設計に当たっては KL1 の並列実行に適したバランスの良いハードウェアと、並列化・分散化オーバヘッドの低い KL1 言語処理系の実現に留意した。この結果、応用プログラムレベルでの工夫の相乗効果により、問題によっては 512 プロセサでほぼ 500 倍近くの速度という高い並列処理効果を得、絶対的な性能としても記号処理システムとして世界最高の速度を得ることができた。

高い総合性能を達成するには、単体プロセサ性能の向上、拡張性の高い階層構造を持つ要素プロセサの結合方式という 2 つの技術が鍵を握っている。KL1 実行に適した高い総合性能を持つアーキテクチャを探るために 5 つのモジュール、PIM/p、PIM/m、PIM/c、PIM/i、PIM/k を開発し、様々な要素技術を検証した(別表参照)。

各 PIM モジュールの仕様

	全 PE 数 [1]	ノード構成			疎結合ネットワーク	
		PE 数	メモリ	バス容量(最大)	トポロジ	バンド幅 [2]
PIM/p	512	8	256MB	130MB/s	6 次元ハイパーキューブ	33MB/s×2
PIM/m	256	1	80MB	40MB/s [3]	2 次元メッシュ(16×16)	8MB/s
PIM/c	256	8	160MB	200MB/s	クロスバ(32×32)	40MB/s
PIM/i	16	8	320MB	30MB/s	---	--- [4]

PIM/k	16	4[5]	1GB[6]	80MB/s(両階層とも)	2階層キャッシュ	---
Multi-PSI	64	1	80MB	25MB/s[3]	2次元メッシュ(8×8)	10MB/s

[1] PE は要素プロセッサ(ProcessingElement)の意。

[2] ノード・ノード間でリンク双方向合わせた値。

[3] メインメモリとキャッシュ間のスループット。

[4] クラスタ同士は SCSI バスで接続。

[5] ミニクラスタは 4PE から成る。

[6] 16PE で共有されている。

	PE アーキテクチャ			
	命令セット	プライベートメモリ [7]	キャッシュ方式 (状態数 [8])	キャッシュサイズ
PIM/p	RISC+ マクロ呼出	IIM(8K 語,1 語=6 バイト), LM(2MB,キャッシュ経路 でアクセス)	スヌープ, 書込無効型(4)	64KB
PIM/m	マイクロプログラ ム	WCS(32K 語,1 語=64 ビッ ト)	書戻型(3)	5KB(i),20KB(d) [9]
PIM/c	マイクロプログラ ム	レジスタファイル (256W), WCS(32K 語,1 語=104 ビ ット)	スヌープ, 書込無効型(5)	80KB
PIM/i	長命令形式 (LIW)	LM: 160K バイト(i),80K バイ ト(d) [9]	スヌープ, 書込更新型(6)	160KB(i),160KB(d) [9]
PIM/k	RISC	LM: 128K バイト(i),128K バイ ト(d) [9]	2階層スヌープ, 書込無効型(4)	1階層 目:128KB(i),256KB(d) [9] 2階層目:1MB(i),4MB(d) [9]
Multi-PSI	マイクロプログラ ム	WCS(16K 語,1 語=53 ビッ ト)	書戻型(3)	20KB

[7] WCS はマイクロプログラムメモリ,IIM は内部命令メモリ,LM はローカルメモリの意。

[8] ロック状態を除く。

[9] (i)は命令,(d)はデータの意

PIM のプロセッサの命令体系には RISC、マイクロプログラム、LIW の 3 種類がある。RISC とは、1 命令で実行する処理の内容を小さく揃え、連続するいくつかの命令を多重処理することで高効率な処理を実現するようなプロセッサアーキテクチャのことである。マイクロプログラムとは、高機能な機械語を、さらにその一段下のハードウェアを直接制御するソフトウェアで記述するようなプロセッサアーキテクチャのことである。LIW とは、1 つの機械語が複数の命令を含むようなプロセッサアーキテクチャのことである。また各 PIM モジュールのプロセッサは、KL1 を効率良く実行するための専用命令やハードウェア(デレファレンス命令、タグアーキテクチャ etc.)を装備しており、その効果も確認できた。

PIM/m を除く各モデルのプロセッサ間結合方式は、負荷の集中を避け拡張性を高めるために階層構造を成している。8 台程度の要素プロセッサ(PE)が 1 本のバスにつながれており、1

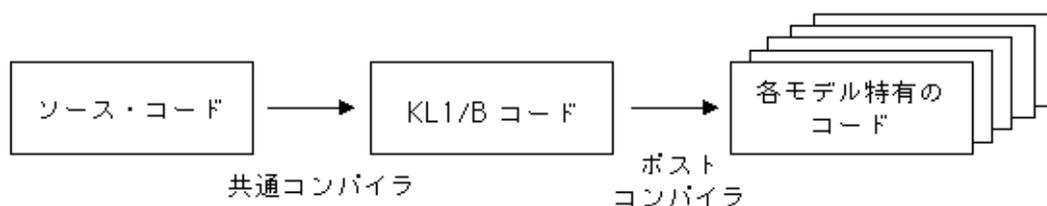
個のメインメモリを共有する。この部分はノードまたはクラスタと呼ばれる。クラスタ内の各 PE はアドレス空間を共有し並列にデータの読み書きを行うので、データの整合性を保ちバストラヒックを抑えるために、スヌープキャッシュという一種のキャッシュが装備されている。さらにこれらクラスタはパケット交換を行うネットワークにより疎結合されている。ネットワークのトポロジにはメッシュ(N 次元格子)、ハイパーキューブ(N 次元超立方体)、クロスバ(完全結合)などがある。ネットワークは、そのデータ転送性能とハードウェア量の兼ね合いを考慮して、KL1 実行に適したものが選択された。PIM/k は疎結合ネットワークを使わずに、キャッシュとバスを階層的に配置して全プロセサを接続するようなアーキテクチャを採っている。

PIM/p と PIM/m の総合性能を概算する。PIM/p、PIM/m の要素プロセサ 1 台は 1 秒当たり 300~600K 回推論を行う能力を持っている。PIM/p 最大構成(512 台)で約 150MLIPS(1 秒間に 1 億 5000 万回の推論を行なう速度)の性能を持ち、これは従来の汎用機に換算すると約 6GIPS に相当する。

並列論理型言語 KL1 処理系

並列論理型言語 KL1 を効率的に並列に実行するための、プログラム言語システムである。同一の言語仕様と共通する処理方式を用いながら、各モデルの PIM に適合した最適化を施している。

特徴的な処理は単一プロセサ内の逐次実行に関する処理と、プロセサ間に渡る処理がある。プロセサ内の処理については、KL1 プログラムをいったん抽象機械語 KL1/B に最適化コンパイルし、これを各モデルのプロセサに適合した方式(機械語へのコンパイルや、マイクロコードによる解釈実行)で実行する方法を取っている。これによって言語仕様の共通化を容易にすると同時に、処理系構築の手間を削減している。



KL1 言語処理系のコンパイル方式

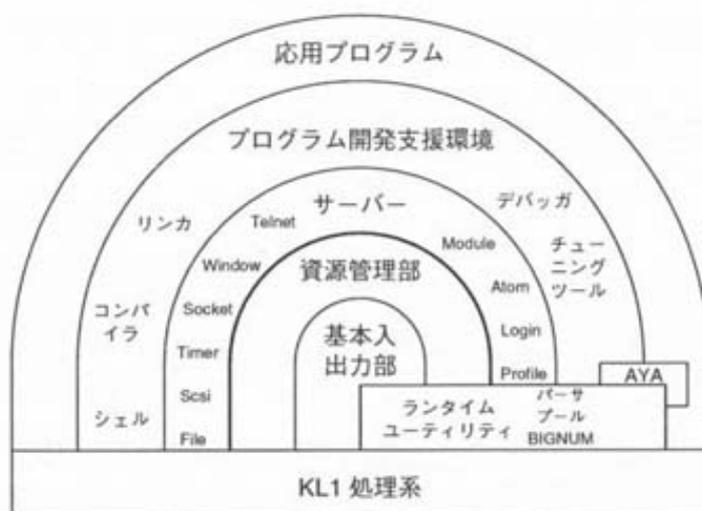
並列処理は共有メモリ・クラスタ内の処理と、クラスタ間に渡る処理に大別される。クラスタ内処理においては、メモリ・アクセスの競合を調停しつつ、自動的な負荷分散を施し、プログラマが細かいレベルでの負荷分散を指定する必要を除去している。一方クラスタ間処理においては、プロセスの分散はプログラム中の指定に従うが、データや実行コードの分配を自動的に行ない、不要になった領域の再利用も自動的に行なう機構を持っている。これらの並列処理機構についても、アルゴリズムはもちろん、具体的なコードも各モ

デルに共通するシステムを用意し、モデルごとに最適な方法でコンパイルなどして実装している。

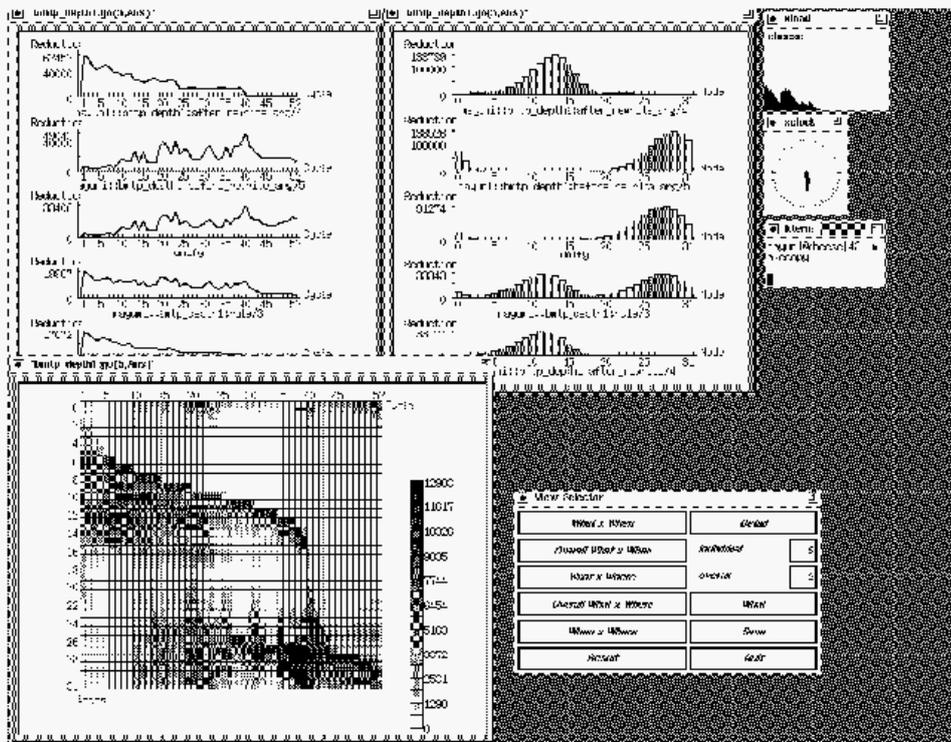
並列推論マシン・オペレーティング・システム PIMOS

並列推論マシンのハードウェアを効率的に管理し、並列プログラムの開発に適したソフトウェア開発環境を提供するオペレーティング・システム。どのモデルの PIM でも同一仕様の PIMOS が動作し、同一のプログラム・インタフェース、同一のソフトウェア開発／実行環境を提供する。

並列処理システム用のオペレーティング・システムは既に数多く開発されているが、大部分のシステムは逐次型計算機用のシステムを若干手直したものであり、プロセサ数十台程度までの小規模システム用か、画一的な処理に向けた SIMD 方式や数値計算などの特殊用途システムでしかなかった。この点 PIMOS は、超高並列システムまでを見通した汎用並列処理システムとして世界で初めて本格的に実装され、実際の研究に役立てられるシステムとなっている。また同時に、世界初の論理型言語による本格的並列オペレーティング・システムである。



PIMOS の全体構成



PIMOS のプロファイラの表示例

高並列システムにおいては、オペレーティング・システムのわずかなオーバーヘッドでも、それが1台のプロセサに集中すれば、システム全体のボトルネックになってしまう。PIMOSは、基本的な資源・実行管理部において徹底した分散管理方針を取ることによって、応用プログラムの実行だけでなく、管理自体も分散処理するようにして、このようなボトルネックが生じるのを防いでいる。

PIMOSは、KL1による並列ソフトウェアの開発に向けたさまざまな機能を提供し、快適なソフトウェア開発環境を提供している。たとえばPIMOSデバッガでは、KL1言語処理系の機能を利用した自動的なデッドロックの検出や、任意のプログラム部分の実行の一時停止・再開、別ウインドウを用いてのプロセス別のトレースや次第に生成されて行くデータの監視などができる。プロファイラでは、プログラム実行中あるいは実行後に、どのような計算がいつ、どこで(どのプロセサで)行なわれているのかをグラフィックに表示し、負荷分散状況の容易な把握を可能にしている。これらの機能を利用することによって、知識プログラミング・システム群や機能実証ソフトウェア群の研究開発は大幅に加速された。

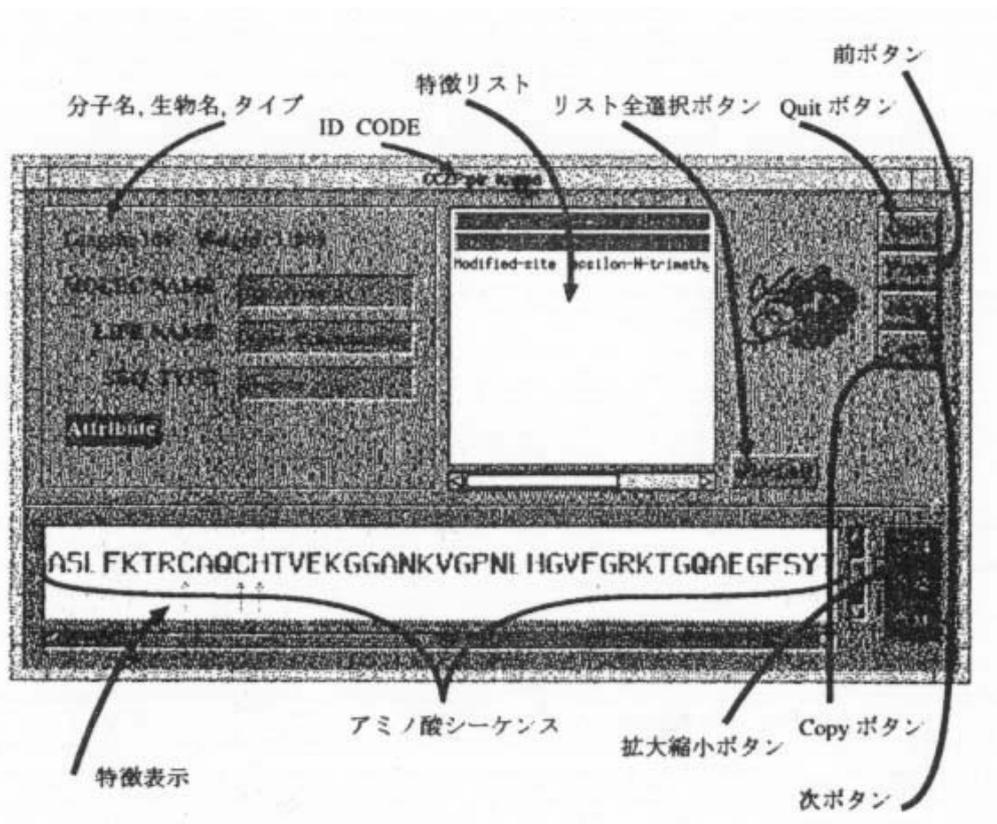
並列非正規関係データベース・システム Kappa-P

並列非正規関係データベース Kappa-P は、知識ベース管理システムや知識情報処理システムや遺伝子情報処理システムに対し、知識ベースを格納するための器としてのデータベース管理機能を提供するものである。

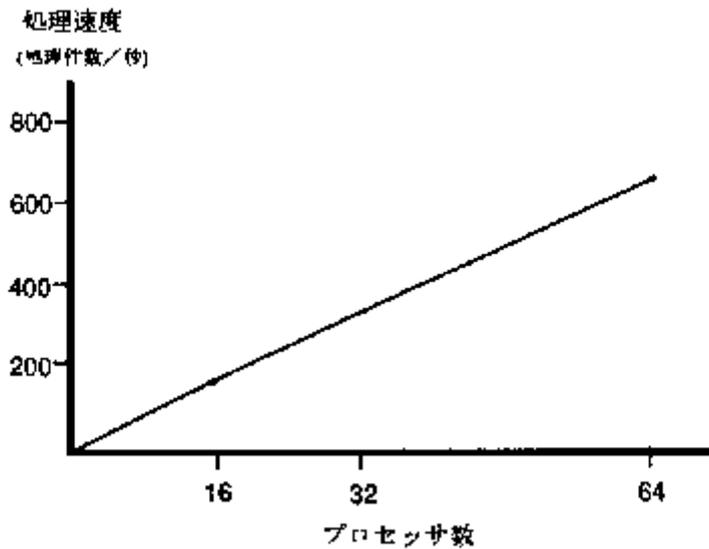
知識ベース管理システムや知識情報処理システムや遺伝子情報処理システムが扱うデータは、構造が複雑かつその内容も多種多様で、それに対する操作もさまざまであり、また、データ量も非常に膨大である。Kappa-P は、そのようなデータを効率的に扱うことができるようになっている。

Kappa-P の特徴は、非正規関係モデルと並列処理にある。非正規関係モデルの採用により複雑な構造のデータを効率的に扱うことが可能になっている。また、並列処理により大量のデータを短時間に処理することが可能になる。

これを確かめるために、分子生物学データベースを統合化した蛋白質データベース(配列情報、三次元構造情報等)を設計しその上で動作する応用プログラムを作った。従来の関係データベース管理システムでは、このような複雑な構造のデータベースを効率的に表現、操作することはできなかった。また、複数のプロセサを使った並列処理により、分子生物学データベースのモチーフ検索などの重い処理を短時間で処理できるようになった。



統合分子生物学データベース・インターフェース



Kappa-P によるモチーフ検索の台数効果

並列定理証明システム MGTP

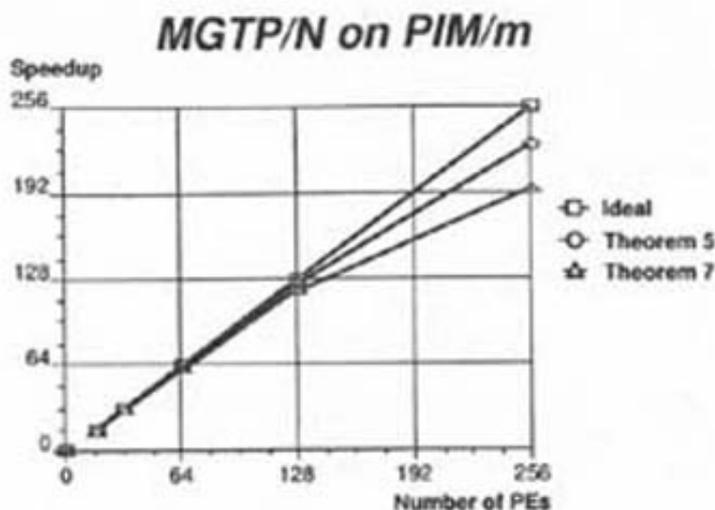
MGTP は高速かつ高機能の自動推論エンジンを提供することを目標に、PIM 上に研究開発されたシステムである。データベース的応用指向及び数学的応用指向の 2 種がある。前者(MGTP/G)は有限領域を、後者(MGTP/N)は無有限領域を扱う定理証明系であるが、両者とも性能に関しては世界最高水準にある。

無限領域の定理証明系としては、これまで米国 ANL で開発された OTTER が機能、性能面ともに他を先行していたが、OTTER 方式に比べ、メモリ量、計算量のオーダを数桁下げる新しい方式(遅延モデル生成法)を開発するとともに、従来困難とされていた分散メモリアーキテクチャ上での並列化方式を実現した。これにより、MGTP/N は PIM/m 単一 PE 上で、SPARCII 上の OTTER の 2~5 倍、PIM/m-256PE 上では、400~1000 倍の性能を達成した。

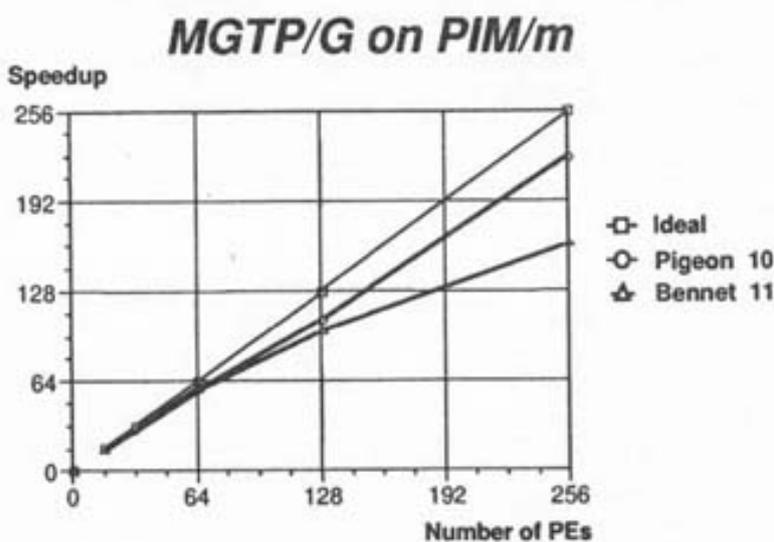
一方、有限領域の定理証明系 MGTP/G については、与えられた問題節を KL1 節に直接コンパイルする手法、並びに場合分けにより生じる組合せ的爆発を抑制する手法(ノンホーン・マジックセット)を開発した。MGTP/G でも、MGTP/N と同様に、PIM/m-256PE 上で 200 倍以上の速度向上を達成している。MGTP/G を用いて、有限代数の問題を試みた結果、カナダの数学者 Benett により提示された準群に関する未解決問題の一部を、自動的に証明することに成功した。

MGTP の機能拡張の一環として、論理プログラミングで開発された「失敗による否定」を MGTP の枠組へ組み込む手法を開発した。本手法は、法的推論システム HELLIHII のルールベース・エンジンに採用されている。さらに、本手法の変形として、仮説推論システムを MGTP 上に構築する方式を開発した。まだ実験段階であるが、従来の方式と比較して

も、遜色のない結果が得られている。



MGTP/N の台数効果



MGTP/G の台数効果

並列制約処理言語システム GDCC

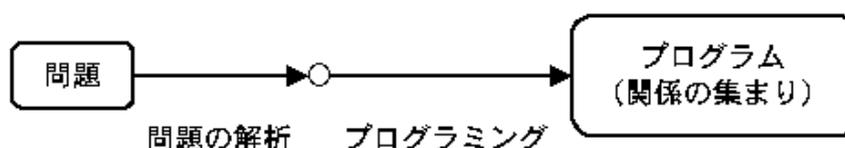
従来のプログラミング言語による問題解決では(1)問題の解析、(2)解法の発見、(3)発見された解法のプログラミングの3ステップを必要としていた。制約論理型言語は、問題をどう解くかではなく、どのような問題を解くかを記述するプログラミング言語である。このため、問題解決を(1)問題の解析、(2)解析された問題のプログラミングという、2ステップだけで行なうことができる。しかも、従来の言語では解法が想定する場面以外に適用するには新たな解法を発見しなければならなかったのに対して、制約論理型言語では特定の場

面を想定せずに、より柔軟なプログラムを記述できる。

従来のプログラミング言語による問題解決



制約論理型言語による問題解決



一方、場面に応じた問題解決をシステムが行うため、システムに要求される機能は増大し、結果的にプログラムの実行効率に問題が生じることがある。特に、制約を解く、制約評価系の処理効率はプログラム全体の効率にとって重要な影響を与えるため、この効率化は不可欠である。その点で並列処理を導入することにより、効率化が実現できた。

並列制約論理型言語 GDCC は、このような効率的かつ柔軟な問題解決のためのプログラミング言語を並列推論マシン上にその処理系を構築することにより、より効率的な実行を目指したものである。GDCC の特徴は、世界で初めて非線形代数方程式を扱える制約論理型言語であり、世界で初めて実際の並列計算機上で動作する並列制約論理型言語である点である。

並列推論マシンを利用することにより、非線形代数方程式を扱うシステムは 16PE で約 6 倍の並列効果が得られ、絶対性能についても、実用的なシステムに必要な性能を達成できた。Vidal による処理系や Siegl による処理系といった、他の並列実装と比較してもより効率の良いシステムが得られ、また効率が高いとされている Giovini による処理系、Backelin による処理系のような他システムと比較しても、複数 PE を利用することで、より高速なシステムを実現することができた(次の表を参照)。これらの比較に際しては、比較的処理時間のかかるベンチマークを選んだため、比較対象となる処理系がすべてのベンチマークを解いているわけではないので、それぞれの処理系でデータが得られているものについての比較を行っている。

ベンチマーク	処理系	PE 数				
		1	2	5	12	16
	GDCC の制約評価系	8	8	4	3	4

Katsura-4 (単位:秒)	Vidal の処理系	17	10	4	4	--
	Giovini の処理系	40	--	--	--	--
Katsura-5 (単位:秒)	GDCC の制約評価系	84	82	27	17	24
	Vidal の処理系	1103	551	146	79	--
Cyc4-roots (単位:秒)	GDCC の制約評価系	1	1	1	1	1
	Siegl の処理系	218	--	--	--	36
Cyc5-roots (単位:秒)	GDCC の制約評価系	23	24	12	11	11
	Giovini の処理系	143	--	--	--	--
T-6 (単位:分)	GDCC の制約評価系	443	453	136	79	66
	Backelin の処理系	90	--	--	--	--

この GDCC を用いて、ハンドリングロボット設計支援システム、階層制約の並列評価システム、ボロノイ図作成支援システムといった応用システムを試作した。それぞれの応用システムにおいて、高い機能と高い効率を実現することができた。たとえば、ハンドリングロボット設計支援システムでは制約論理型言語の持つ柔軟性を生かした、高機能で高効率な支援系を実現し、また、ボロノイ図作成支援システムでは、その定義を直接プログラムとして記述することで、高機能で、ほぼ PE 台数に比例する台数効果が出るという、高効率のシステムが得られた。

知識表現言語システム Quixote

より大規模で実用的な知識情報処理システムを実現するためには、大量で複雑な知識を計算機上で効率良く扱う機能が必要がある。これが知識ベース管理機能である。これを実現するためには以下のことを考慮する必要がある。

1. 複雑で雑多な知識をいかに表現するか
2. 大量の知識をいかに効率良く蓄積するか
3. 蓄積した知識から必要な知識をいかに導出するか

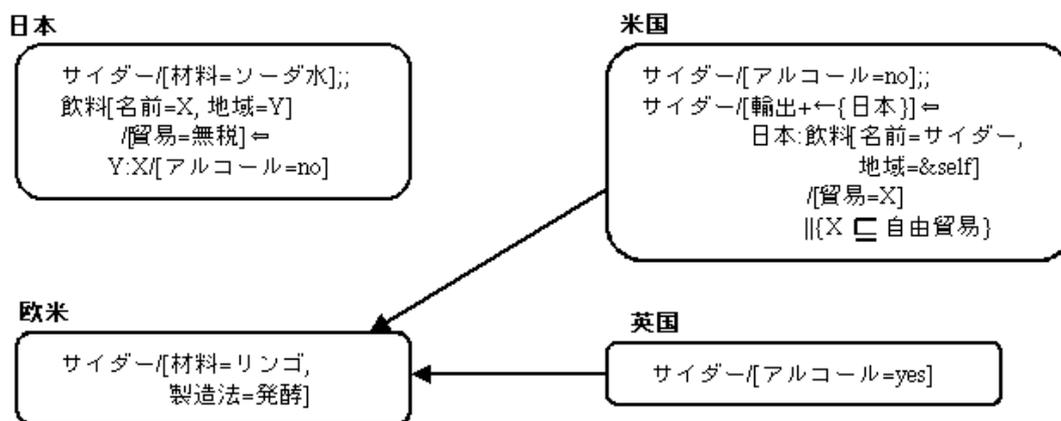
このために、知識表現言語 Quixote を設計し、それによって表現された知識を格納した知識ベースを管理し、必要な知識を導出するためのシステムを開発した(このシステムを Quixote システムと呼ぶ)。

Quixote 言語は、実世界の知識を自然に表現するために、データベース、プログラミング言語、自然言語処理などの複数の分野から検討を加えて設計されたマルチパラダイム言語である。このような知識表現には、オブジェクトの情報は完全であるか不完全であるかによって従来複数の流れがあったが、Quixote はこれらを自然に統合した言語としては世界で初めての言語となっている。

Quixote システムは、上記の知識ベースに対する高度な利用を可能にするために、推論機能を強化し、大量の知識をディスクに格納する機能をサポートしている。前者の機能としては、通常の演繹の他に、知識ベースに仮説を追加して推論する機能や、知識ベース

に不足している知識を推論する機能などもサポートしている。このようは包括的な機能を持った知識ベース・システムは他に例を見ない。

この Quioxte 言語と Quixote システムの有効性は、新しい事件の判決を予測する法的推論、複数の自然言語表現から曖昧性を除去する状況推論、複雑な構造体間の関連を推論する遺伝子情報処理などの応用によって確かめられている。



Quixote で知識がいかにか表現されるかを例で示す。たとえば

1. サイダーは、日本ではソーダ水の種類だが、欧米ではリンゴから発酵によって作られたものである。
2. 米国のサイダーにはアルコールは含まれていないが、英国のサイダーにはアルコールが含まれている。
3. 日本では、アルコールの含まれていない飲料については無税である。
4. 日本のサイダー市場が開放されているならば、米国は日本にサイダーを輸出する。

のような知識は図に示すように書くことができる(図で示しているが、Quixote による記述量もほぼ同じになる)。

法的推論システム HELIC-II

HELIC-II は、新しい事件を与えると、それに対してさまざまな法律適用の可能性を提示するシステムである。一般に、法律に書かれているルールは抽象的なので、与えられた事件にルールをダイレクトに適用することはできない。そこで、HELIC-II はまず与えられた事件と類似する判例を探し、そこで用いられている法適用の論理を利用して抽象的な概念仮説を作りだし、その後でルールを適用する。このシステムは、判例の並列検索と条文の並列適用によって高速な推論を可能にしている。この HELIC-II は、刑法の条文と刑法の判例を格納して、刑法の実問題をシステムに解かせ、その問題解決能力を分析している。このシステムの技術的評価を以下に述べる。

条文による推論と判例による推論を統合した新しい推論メカニズム

2つの推論モジュールを組み合わせるにより、ルールによる推論の厳密さと事例による推論の柔軟さという利点を活かした推論システムを実現している。なお、ルールによる推論のモジュールは、並列定理証明プログラム MGTP を拡張したものである。また HELIC-II は、事例による推論を初めとする強力な推論方式を実現している。従来、強力な推論手法は時間がかかることが欠点であるとされてきたが、これらを並列化することにより、実用的な時間で結論が求まることを示した。例えば、類似の判例を検出するモジュールでは、PIM の1台のプロセサでは2時間以上かかる計算が、64台では3分程度で実行できている。

詳細な知識表現と高度な推論技術

刑事事件をコンピュータで扱うには、当事者の認識と事実の食い違い、当事者の意図の時間的変化、1つの行為の立場による解釈の相違、などの詳細な情報を記号として表現することが必要とされる。また、それらの情報を利用するには、類推や仮説の生成や因果関係や時間関係などの推論技術が必要とされる。このシステムでは、刑事事件の実問題の解決に必要な知識表現や推論の手法が開発された。知識表現については、さらに状況理論を用いて形式化を行い、推論技術については、さらに計算モデルとして形式化を行った。その結果、法的推論がどのような性質を持つ推論であるかを理論的に解析することが容易になった。知識表現や推論の要素技術は未だに発展段階にあるため、汎用で強力な知識表現/推論のシステムを開発できる状況にはない。本研究では、法的推論という枠組の中で、個々の要素技術を有機的に統合して強力な推論機構を実現している。しかも、一般性を持たせるために、法律の特殊性をなるべく排除するように工夫されている。また、入力を容易にするためのインタフェース、自然言語による出力インタフェースなども用意されている。

実問題を解決することにより推論能力を実証

上記の推論メカニズムの有効性を示すため、司法試験の事例問題を解くための知識を分析し、PIM の上で実際に試験問題のいくつかが解けることを示した。試験問題の解答とシステムの生成した解答を比較し、このシステムの持つメカニズムで現実の多くの問題が解決できることを示した。他の法的推論システムの多くは単純化した事件のみを例題として用いていたが、本システムでは、実際の判例を分析して論理の流れを抽出している。その際、標準的な教科書で議論されている学説レベルの詳細度の問題分析を行っている。

高速な推論を可能にした並列プログラミング技法

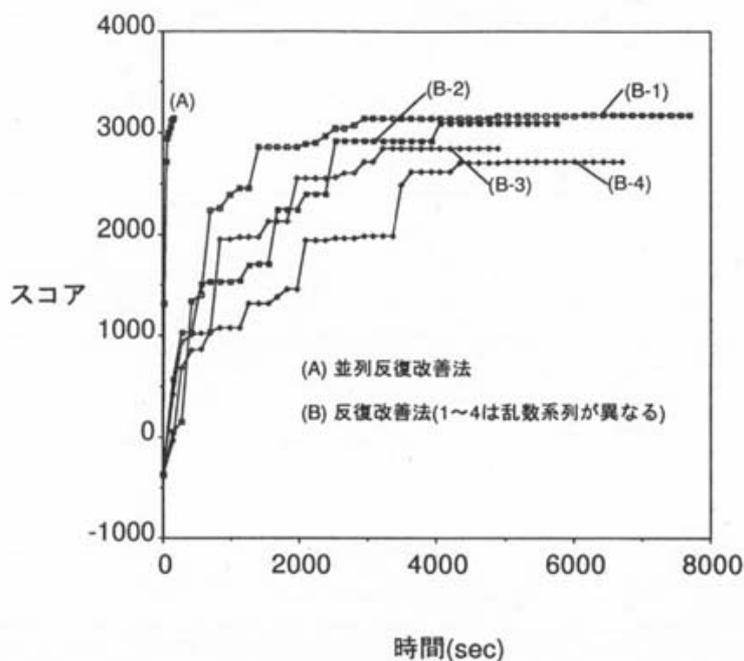
推論システムを効率的に動作させるために、並列プログラミング技法が開発された。

例えば、2つの事件の類似性を高速に判定する方法、2つの推論モジュールを並列に走らせる方法、仮説の組合せが不必要に生成されないようにする方法などのプログラミング技法が開発された。

システムの推論モジュール部分（第1版）は並列プログラミング言語 KL1 で約 5,000 行と、従来技術による場合に比してひと桁以上コンパクトな記述、およそ9人月という少ない工数で開発されている。これは、複雑な並列応用システムに対する KL1 の記述力を実証する好例となっている。

遺伝子情報処理システム

タンパク質配列の類似性解析技術は、タンパク質の分類、未知の機能の推測、進化の過程の解明に重要な役割を担う技術である。従来は、もっぱら熟練した生物学者が経験と勘を頼りに時間をかけて行っていたが、近年、解析すべきタンパク質配列の数が急激に増加しており、解析技術の計算機による自動化が望まれていた。以前より、いくつかの類似性解析手法が提案され、計算機に実装されてはいるが、生物学的な観点からすると解の品質が不十分であったり、満足すべき解が得られるまで許容できないほどの長時間を要するものが、ほとんどであった。



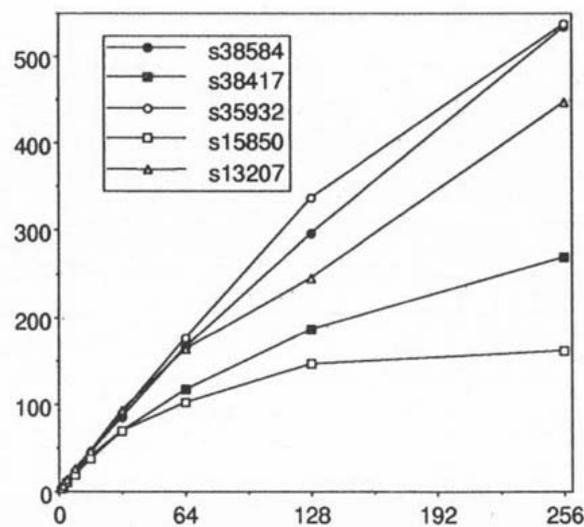
並列反復改善法の効果

このシステムは、並列推論マシンを利用し、実用規模の類似性解析を現実的な時間内に実行し、高品質な解析結果を与えるシステムを構築している。本システムは独自に開発した並列反復改善法を用いており、部分的な最適化を並列に、そして反復的に行っている。

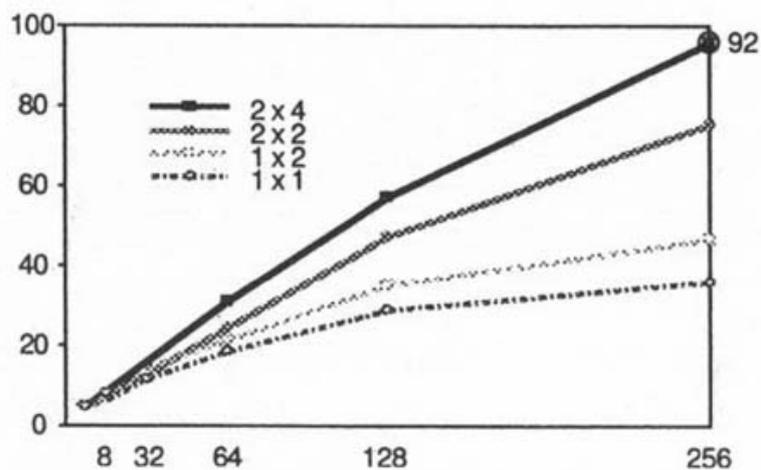
具体的には、反復の各段階で、可能な部分的最適化を複数、それぞれ各要素プロセサで並列に行い、そのうちで一番評価の良い結果を採用していく方法をとっている。この方法により、乱数で最適化の手順を決め、逐次的に改善を行う反復改善法よりも、256 台構成の並列推論マシンで、およそ 85 倍の高速化を得ている。

大規模集積回路設計支援システム

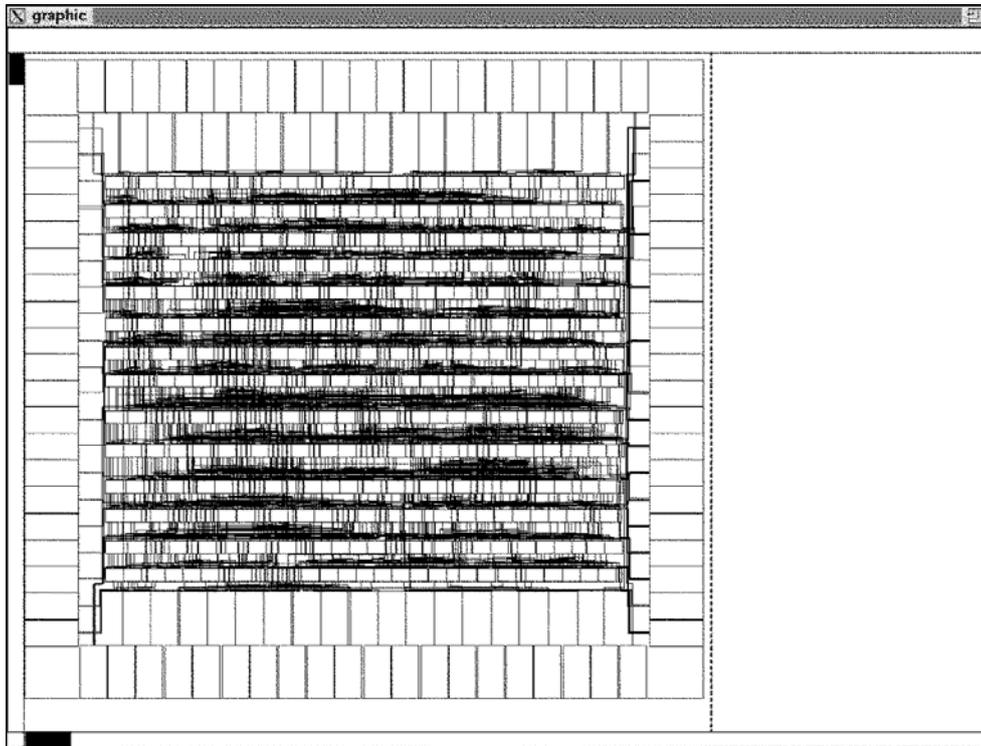
LSI-CAD の工程中、論理シミュレーション、LSI セル配置、および LSI 配線の工程を並列化し、高速化および高品質化を達成した。



論理シミュレータの台数効果
(縦軸:K イベント/秒;横軸:プロセッサ数)



LSI 配線の台数効果
(縦軸:台数効果;横軸:プロセッサ数)



LSI 配置・配線の結果表示

論理シミュレーション、LSI 配線は数日～数か月という多大な処理時間を必要とする工程であり高速化が強く望まれている。また、LSI セル配置では、人手による微妙なパラメータ設定が必要であり、適切な結果を得られない場合が存在する。これらの問題点を解決するために、このシステムでは下記の技術を開発、導入している。

論理シミュレーション

見込み計算(無駄になるかも知れない計算)を積極的に行う機構を導入することで、多大な並列性抽出に成功。また、プロセッサ間通信を低減しつつ、負荷の均等化を達成する回路分割アルゴリズムを提案。166 倍の並列処理効果(256 プロセッサ使用時)、数百 MIPS のスーパーコン上のシステム相当の性能を達成。

LSI セル配置

適切なセル配置を求めるためにシミュレーテッド・アニーリング法(金属の焼き鈍し工程を模擬する方法)を採用し、これにパラメータ設定を自動化する並列アルゴリズムを導入。これによって、極めて最適に近い配置を得る能力を実現。既存のシステムによる結果と比較し、最小のチップ面積を達成。

LSI 配線

オブジェクト指向のモデル化技法を導入することによって、多大な並列性抽出に成

功。また、並列に配線経路探索を行いつつ、指定された配線順序を守ることで、配線品質を向上させる機構を提案。92 倍の台数効果(256 プロセッサ使用時)、数百 MIPS スーパーコン上のシステム相当の性能を達成。また、100%の配線率を達成。

自然言語処理システム

概要

第五世代プロジェクトの中期頃までは、適当な例文について人間と機械との対話を行なうシステムを試作した。この結果、人間と機械との対話を円滑に行うためには、従来の文法で扱っている文内の構造のみならず、複数の文にまたがる文脈構造の解析・生成規則が必要なこと、また言語的知識だけでは不十分であり、対象分野やその周辺に関する膨大な背景知識が必要であることが明白になった。その結果、限られた例文の中でトイ・システムの全体を通じて動かすことには重点を置かず、個々の要素技術、個別のテーマをさらに深めて研究し、得られた成果を広く世界に公開する方針を取った。

文脈構造の解析・生成については、従来の構文文法の拡充・整備を継続する一方で、いわば文脈文法ともいべき規則の解析収集を行った。背景知識については、膨大な言語外知識の収集は ICOT における作業では限界があり、むしろ日本電子化辞書研究所 EDR で行われている大規模な辞書、概念辞書の開発、あるいはアメリカの MCC における百科事典的知識の収集、記述プロジェクト CYC などの成果に期待する方針を取っている。

言語データの蓄積

以上の方針のもとで、文法・辞書を対象に言語データの収集、整備を行った。文法に関しては論理型プログラミングにおける単一化文法の延長上にある句構造文法の枠組みと、従来から特に日本語を対象として適用されてきた係り受け文法の枠組とにしたがって日本語文法を試作した。新しい言語現象に出会うたびに文法を修正していく方針では、文法が大きくなるにつれ体系全体の整合を取ることが困難になるので、日本語の言語現象を分類し、それぞれの現象を押える文法規則を網羅的に記述する方針をとった。この文法に関しては、日本国内のみならず、近年日本語処理に対する興味が高まってきている諸外国の研究機関からの利用希望も数多く寄せられた。

このような文法規則の開発にあたっては、支援ツールの整備が必須なので、作成中の文法を実際に利用して解析を行い、評価、修正を容易にする文法開発支援環境 Linguist や、現実のテキストから指定された言語現象を抽出する KWIC などのツール類を試作した。

また、文脈規則に関しては、新聞社説の文章を対象とし、隣接する文の間関係や文中の語の間関係から文脈構造を抽出する実験を行った。この結果からはいわば文脈文法に相当する文脈解析規則が収集された。

辞書に関しては、形態素情報を中心にした語彙辞書を整備し、約 10 万語、15 万形態素程度の情報を蓄積し、多くのシステムで利用された。

汎用日本語処理系のツール群の公開

中期に試作したツール群はほとんどが ESP で書かれており、PSI 以外のマシン上で稼働させることはできなかった。そこで、AI 言語研究所 AIR において開発された CESP(CommonESP)に移植することによって、一般の UNIX マシン上で利用可能なものとし、多くのユーザに公開した。CESP に移植したツールとしては、形態素解析システム LAX、文法開発支援環境 Linguist、文生成ツール、KWIC などがある。さらに、いくつかの逐次、並列構文解析ツールに関しても、C、Prolog、KL1 の形態で公開した。

文生成ツールの拡張

中期までに表層の文にきわめて近い内部データから出発し、それを日本語の文字列に変換する文生成ツールを試作した。しかし、実際にインタフェースとしてこのツールを使い易くするためには、聞き手の内部状態、信念を勘案し、発話すべき内容を組み立てる処理が必要であることがわかった。そこで、発話のためのプランニングを伴う文生成ツールの試作を行った。

このツールは、事実、ルール、判断などを含むシステムの信念に対し、立論ゴールを与えてプランニングを起動する。その結果、聞き手に伝えたい内容を、いわば説得術に相当する言語的・修辭的な戦略に沿って整理して組み上げ、さらに、自分の主張だけでなく、それを補強するための例示、予想される反駁、反駁に対する反証なども表現する。本ツールは並列マシン上のプログラムとして試作された。

プログラム自動合成システム MENDELSZONE

MENDELSZONE は形式手法を用いて並列プログラムの開発を支援するシステムである。従来より、ソフトウェアの高信頼性、高生産性を得るために、形式手法が有効であるとの指摘があった。形式手法を用いることで、ソフトウェア仕様の自動検証や調整、仕様からプログラムへの自動変換が可能となるためである。しかし、現実には(1)検証・調整に要する多大な計算時間、(2)変換されるプログラムの実行効率の低さが障害となって、実開発への適用は実質的に困難であると考えられていた。この問題に対して MENDELSZONE では、

(1)すべての検証・調整手続きを並列化することで計算時間を大幅に短縮、

(2)KL1 プログラムを生成することでプログラムの実行効率を格段に向上

させ、第五世代コンピュータの高い計算能力のもとで、形式手法によるソフトウェア開発を世界で初めて現実のものへと近づけた。これまでの実験では、検証について並列要素数に対してほぼ線形に性能が向上し、生成されるプログラムについては、宣言的な記

述から得られるプログラムであっても従来の 10 倍以上の速度で実行されることが確認されている。

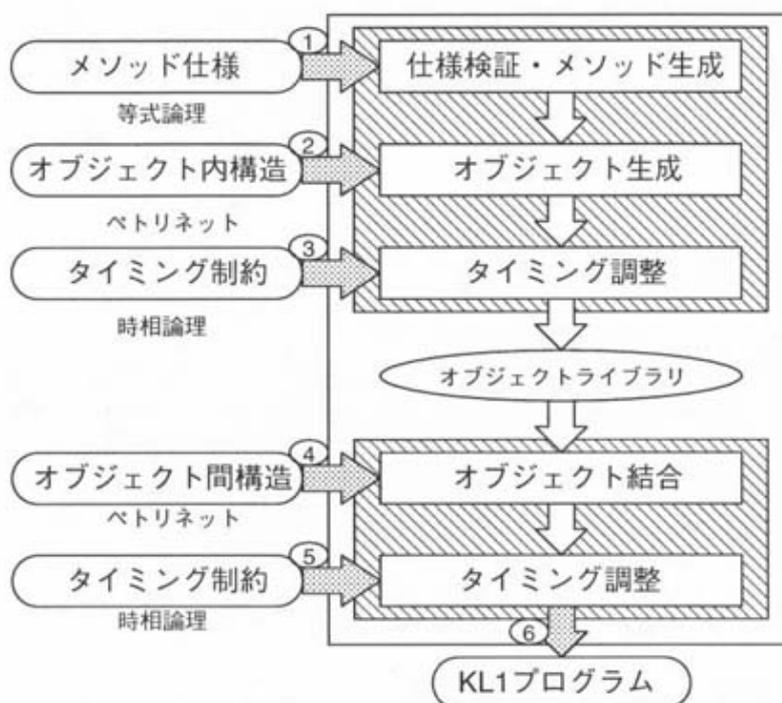
また、形式手法によって実システムを記述する際の問題点として(3)システムのさまざまな性質を単一の手法によって記述する難しさ、(4)大規模システムを形式記述する手間の多さ、(5)形式記述の理解性の低さなどが挙げられる。これらについては、

(3)複数の形式手法を用意し、記述する性質に適した手法を選択、

(4)形式手法にオブジェクトの概念を導入し、オブジェクト単位の部品化・再利用を実現、

(5)形式記述を図形表示し、記述の理解性を向上

させ、記述性の問題にも対処している。複数の形式手法には、機能を記述するための等式論理と、タイミングを記述するための時相論理、構造を記述・表示するためのペトリネットが用いられる。ペトリネットは、プログラム実行の様子をモニタリングする際にも活用される。



MENDELSZONE におけるソフトウェア開発の流れ

図に MENDELSZONE におけるソフトウェア開発の流れを示す。(1)では等式論理による機能(メソッド)仕様が入力され、その正しさが並列に自動検証される。(2)ではそれらの構造がペトリネット表示される。(3)では時相論理による機能間のタイミング仕様が入力され、その仕様を満足するように(2)のペトリネットが並列に自動調整される。(4)では複数のオブジェクトがペトリネットエディタを用いて結合され、(5)ではオブジェクト間の

タイミング仕様が入力され、(4)のペトリネットが自動調整される。最後に(6)でこれらの仕様が KL1 プログラムへ変換される。

MENDELSZONE の評価を目的としてプラント制御エキスパートシステムを開発した結果、約 4,000 行の形式記述から 6,200 行の KL1 プログラムが生成され、約 5 人月を要して目的のシステムが完成した。過去に同一システムを手作業により開発した際のデータと比較すると、プログラムサイズは約 34%増加しているが、開発者が記述するコードの量は約 15%減少している。開発工数は従来の約 1/2 で、特にデバッグ時間が 1/12 となっている。これにより、実開発における MENDELSZONE の有効性が確認された。

棋士システム碁世代

囲碁の対局を行なう棋士システム。PSI 上の逐次版と PIM 上の並列版とがある。

コンピュータ囲碁は、チェスなどに代表されるコンピュータゲームプログラムで成功した探索主体の方法では強くなれない。そこで、人間プレイヤーの思考方法のシミュレーションを通じて「碁世代」を開発した。局面認識の方法や、次の着手を考える際に盤面の重要なところに絞り込んで考える点に特徴がある。逐次版「碁世代」はアマチュア中級程度の強さを持ち、1992 年の世界コンピュータ囲碁選手権で 4 位となった。

並列版「碁世代」は、世界初の並列処理を取り入れた囲碁システムである。並列版「碁世代」では、次の一着の決定は一定の時間内に収めながら、暇なプロセッサに後で役に立ちそうな別な箇所を検討させる方式を考案・試作し、さらに強いシステムを作れることを実証した。