

29-30

## FGCS 的后续计划

TP338.6

刘海燕 张 静 编译 王献昌 审校

## 1. 引言

为期十年的 FGCS 计划结束了, ICOT 取得了令人瞩目的研究成果, 并在 FGCS'92 会上展出。成果核心是 FGCS 原型系统, 它包括并行推理机 PIMS(有1000个 PE, 速度达100MLIPS)。在 PIM 上演示了许多并行软件系统, PIMOS、Kappa-P、Quixote、GDCC、MGTP 和20个并行应用系统, 其中法律推理系统和遗传信息分析系统倍受关注。成功的演示证明知识的并行处理行之有效, 某些应用系统的 N 个 PE 可达到几乎 N 倍的加速! 如定理证明器。总之, FGCS 的科研成果得到了逻辑程序设计和并行符号处理技术领域专家的高度评价。

然而, 新应用和传统计算机技术领域的人们提出了一些非议和新的要求, 有人认为 FGCS 的软件产品应移植到目前市场流行的计算机系统上, 不仅限于 PIM 上的应用; 也有人要求努力开发知识处理应用。这些批评和要求在十年计划期间很难处理妥当。

日本基础计算机技术开发委员会收集国内外数以百计的新计算机技术人员的观点, 经近半年的讨论, 决定建议 MITI 开展一个为期两年的后续计划, 1993年四月 MITI 和 ICOT 开始了实施。

后续计划的目的是把 FGCS 计划中所开发的主要软件系统和基础研究成果做为开发高级计算机技术(核心是并行知识处理)的工具和支撑环境, 广泛应用于研究团体, 具体目标是:

- 1) 把 KL1、PIMOS 和主要软件系统移植到基于 UNIX 的顺序系统和 MIMD 并行系统上。
- 2) 采用 PIMS 和 KL1 实现的软件系统, 进一步

循环结构都可表示为对象标识符。

## 6. 结论

本文总结了 FGCS 计划的基本研究活动, 强调了逻辑程序设计研究的两个不同方向: ①逻辑程序设计语言。焦点是约束逻辑程序设计和并发逻辑程序设计。②AI 的基础研究和基于逻辑、逻辑程序设计的软件工程。

研究开发知识处理软件和并行应用软件。

目前, 许多并行 MIMD 机进入市场, 为 KL1 和 PIMOS 的移植提供了合适的体系结构和性能并且适合于 ICOT 开发的多种并行应用。移植为 KL1 软件和基于 Unix 的系统之间提供一个有效的软件界面, 同时 KL1 也提供了一个非常高效的符号和知识处理的并行程序设计环境。

ICOT 预备精简机构, 7个实验室和规划部(近100人)减至2个研究部门和1个规划部(近40人)。总务部和国际关系部将不变(约10人), ICOT 人员更替, 但大多数研究组长将继续工作两年以确保研究活动继续, 为传播 ICOT 自由软件(IFS)和进一步开发并行知识处理技术, 继续寻求国际合作。

## 2. 把并行推理系统移植到库存硬件和 OS 上

后续计划将开发一个可移植的、高效实现的 KL1 语言及其软件开发环境并把它们作为 IFS 公开发表, 而已作为 IFS 发表的软件系统将移植到基于 Unix 的系统上。

## 2.1 Unix 和 C 作为平台

由于流行 Unix 操作系统和 C 语言, 所以被选择为目标系统。遗憾的是, C 语言用于 KL1 的实现并不理想, 而且 Unix 对于并行语言的实现也不尽人意, 因为它的标准内部进程通讯机制相当弱。但考虑到可移植性, 仍基本坚持 Unix 特征的一个非常标准的子集。尽管将可移植性放在第一位, 但缺乏高效的系统是没有价值的。采用合适的目标代码设计可以合理地掩盖 C 语言作为一种语言实现工具的不足, 通过把 KL1 代码编译成 C 来完成一个小子集的顺序实现。在 Sparc Station 2 上, 这种实现每秒能减少

ICOT 将在以下领域继续努力。①约束逻辑程序设计和并发逻辑程序设计的结合; ②用 KL1 实现一个数据库; ③追求诱导和归纳推理的并行实现。

主要参考文献: Koichi Furukawa, Summary of Basic Research Activities of the FGCS Project, Proceedings of the International Conference on Fifth Generation Computer Systems 1992, ICOT

约一百万个 KL1 目标,近似于 PIM PE 的执行性能。

## 2.2 目标体系结构

计划采用三种不同的 Unix 系统做硬件平台。

**顺序系统:**顺序 Unix 系统,如廉价的个人工作站,将有效地用于软件开发阶段早期。

**多处理机系统:**基于 Unix 的多处理机系统将提供高性能,并超越 PIM 系统。

**互连网络系统:**顺序 Unix 系统与标准以太网连接可提供合理性能而无需太多硬件开销。多处理机系统与高性能网络的连接将获得最大执行性能。

## 2.3 语言

KL1 语言规格说明基本保持不变,一些元级程序设计特征,如 shoen 结构将稍加改变以适应 Unix 系统。计划提供两类不同的外部语言界面 (foreign language interface):一是常见于许多程序设计语言实现中的子程序界面;二是进程界面,其中的外部语言进程(非 Unix 进程,而是很象面向对象并发语言的对象的细粒度 KL1 进程)可在 KL1 系统中保持它们的内部状态而无需转化为 KL1 实现的内部表示。

## 2.4 规划表

·计划于 93 年四月发表一评估系统以评估其实现模式。由于缺少调试辅助,只由语言实现专家进行评估,不面向用户。

·期望于 93 年 9 月为应用用户开发第一个顺序系统,为调试和性能分析提供一个合理的软件开发环境,同时公布一些应用程序库。

·预计 94 年第二季度公布第一个并行系统。

## 3. 发展知识处理技术

后续计划的另一主要任务是开发大规模知识处理技术。ICOT 于 FGCS 计划期间在 PIM 上广泛开发了知识处理技术(如知识表示语言 Quixote; CLP 语言 CAL 和 GDCC; 定理证明器 MGTP)和应用系统(如遗传分析系统、法律推理系统、LSI-CAD 系统和博弈系统)。在后续计划中,将继续各项研究并把各种技术结合起来。

### 3.1 Quixote

Quixote 是基于逻辑和面向对象风范的知识表示语言和基于类属关系的约束逻辑程序设计语言。从数据库观点看,它是一个演绎的面向对象数据库 (DOOD) 语言,从环境理论观点看,它又是一个环境程序设计语言。在 FGCS 计划期间设计实现了 Quixote 并用它构造了一些实验性知识库,遗传和法律信息系统。后续计划将扩展 Quixote 以更好地应

用于知识信息处理;嵌入否定信息和反意 (disjunctive) 信息、多知识库协作问题求解,Quixote 平台语言设计。

### 3.2 CsLP (约束逻辑程序设计)

CsLP 是扩展逻辑程序设计后得到的一种新程序设计范型。FGCS 计划期间实现了一个顺序 CsLP 语言 CAL 和一个基于提交-选择 (committed-choice) 的并行 CsLP 语言 GDCC,这两者都能处理各种领域的约束,包括用 Buchberger 算法解代数非线性方程,而 GDCC 的并行版本可有效地使算法并行化。后续计划旨在刷新 GDCC 的功能和效率,即改进语言规格说明以达到更加灵活的控制结构,提高 Buchberger 算法并行实现的效率,更进一步实现能处理各种约束的并行系统。通过这些研究活动,目的是创建强有力的并行 CsLP 语言。

### 3.3 定理证明器

FGCS 计划期间用 KL1 开发的并行定理证明器 MGTP 目前有两个成员:MGTP/G 和 MGTP/N。前者处理基础的非 Horn 问题 (ground non-Horn problem); 后者处理非基础 Horn 问题。用 MGTP 可使 256-PEPIM 上的基准程序获得几乎线性加速。MGTP 已投入 AI 的一些应用中,如诱导推理、法律推理和自然语言分析。后续计划将进一步提高 MGTP 的效率并将其扩展应用于反意 (disjunctive) 数据库。目标是通过研究开发 KL1 编译技术、有效证明的元控制方法和统一自顶向下和自底向上计算的非 Horn 魔集来提高 MGTP 到一个新的自底向上程序设计语言系统。

### 3.4 遗传分析

FGCS 计划为遗传分析进行了三个小项目:第一个是并行多序列排队器,它通过比较氨基酸序列来抽取蛋白质特征;第二个是蛋白质折叠模拟器用于模拟水中的每种氨基酸的性状;第三个则用 Quixote 描述了生物学现象。后续计划将试图通过引入人类专家的知识来完善多序列排队器,建立电子邮件服务以接收来自生物学家的初始序列数据并返回调整数据,预言蛋白质的更高阶结构。

### 3.5 法律推理

FGCS 计划开发了并行法律推理系统 HELIC-I,它包括一个基于规则的引擎 (engine) 和一个基于案例的引擎,能通过参考历史案例,根据罚金代码来解决犯罪问题。后续计划将开发一个软件工具,其中包括一个解释模块,用于开发各种法律推理系统。研究将寻求与 AI 有关领域(如各种高阶推理机制、两个原告代理与被告的辩论机制)的科研人员的合作。  
主要参考文献,Shunichi Uchida et al., Outline of the FGCS Follow-on Project, New Generation Computing, 11(1993)217-222