

5

19-22

可移动智能代理系统*

王杰 白英彩

(上海交通大学金桥网络工程中心 上海 200030)

计算机网络
分布式系统
可移动代理系统

摘要 The paper presents basic principle of Mobile Intelligent Agent (MIA) operation, working-mode (i. e. "do-jump" and "jump-do" mode), Status Transmission Graph (STG), layered architecture (i. e. Transport layer, Cooperation layer, Agent layer, and High level layer) and some key technologies in designing MIA, and discusses advantages of Mobile Intelligent Agent Computing Model over traditional Client-Server Model. Finally, we discuss several practical MIA systems and future work on MIA research.

关键词 Mobile Intelligent Agent (MIA), Distributed AI (DAI), Multi-agent system (MAS), Client-Server Model, Intelligent Agent

1 引言

计算机网络和分布式人工智能技术的高速发展,推动了新型分布计算机理论的研究与发展,特别是在 WAN 广泛建立与使用的今天,基于客户机-服务器(Client-Server)模式的超大规模分布计算和可移动计算有了很大的发展.这些方法对于数据量(用户量)较小情况还是十分有效的,但对于数据量较

大、网络带宽较窄、且延迟较大的情况,就比较难适用.究其原因是客户机与服务器之间大量的中间计算结果传输浪费网络带宽,使得分布式计算的效率较低;另一方面,随着客户机数目和要求功能的增加,服务器控制愈加复杂,因而其可维护性和扩充性方面较差.可移动代理技术应运而生^[1,2],采用可移动代理来补充传统客户机-服务器模型是十分有意义的.90年代初 General Magic 公司推出了可移动代理系统

2. 掷色子问题:掷一次色子,正面值 r 可取得 1, 2, 3, 4, 5, 6. 我们计算这样一个事件的概率: "[if ($3 \leq r \leq 5$), then r 是奇数] or [if ($4 \leq r$), then r 是偶数] and [if r 是偶数, then $r \leq 4$]" 用条件事件可表示为:

$$[(\text{odd} | 3 \leq r \leq 5) \vee (\text{even} | 4 \leq r)] \wedge (r \leq 4 | \text{even})]$$

$$\text{解: 首先 } [(\text{odd} | 3 \leq r \leq 5) \vee (\text{even} | 4 \leq r)] = [(\text{odd})(3 \leq r \leq 5) \vee (\text{even})(4 \leq r)] | [(3 \leq r \leq 5) \vee (4 \leq r)] = [(3 \vee 5) \vee (4 \vee 6)] | (3 \leq r) = [(3 \leq r) | (3 \leq r)]$$

$$\text{其次 } [(3 \leq r) | (3 \leq r)] \wedge [(r \leq 4) | \text{even}] = [(3 \leq r) \vee (3 \leq r)'] | [(r \leq 4) \vee (\text{even})'] | [(3 \leq r) \vee \text{even}] = (r \leq 5) | (r \leq 2) = (2 \leq r \leq 5) | (2 \leq r)$$

所以上述事件的概率为 4/5.

结束语 条件事件代数是以前智能系统中的不确定推理为背景而建立的,所以它的许多结果可直接应用于专家系统的规则设计与来自不同信息源的信息的

融合. 本文所举两例仅是为了说明条件事件代数在条件命题合成时的作用,以激发人们进一步研究的兴趣,另外,条件事件代数目前还不是一个完善的理论,有许多问题有待解决,如高阶条件事件的表示与运算,条件随机变量的定义,条件事件代数与模糊推理的关系都是很有意义的工作,我们期望有更多的同行从事这方面的研究工作.

参考文献

- [1] I. R. Goodman, Toward a Comprehensive Theory of Linguistic and Probabilistic Evidence: Two new approaches to conditional event algebra, IEEE Trans. on Systems, Man and Cybernetics, 24(12)1994
- [2] Philip Calabrese, An Algebraic Synthesis of The Foundation of Logic and Probability, Information Sciences, (42):987
- [3] I. R. Goodman et al., Conditional inference and logic in intelligent systems: A Theory of Measure-Free Conditioning, Amsterdam, North-Holland, 1991

* 本文得到中美合作项目支持. 王杰 博士生, 目前主要研究为计算机网络、CSCW 和智能代理网络系统. 白英彩 教授, 博士生导师, 主要研究兴趣为计算机网络、分布式系统理论及其应用.

的商业版本 Telescript, Sun Microsystem 公司研制 Java 以及 IBM, AT&T, Dartmouth 学院等均开始了可移动智能代理的研究。目前可移动代理的主要应用领域包括: 通讯网络管理、分布式信息检索、 workflow 管理等。

2 可移动代理系统

可移动代理的基本思想起源于“移动机器人”, 因此可移动代理也可看成为“可移动的软机器人”, Etzioni 和 Weld 曾提出在 Internet 上把 FTP, Telnet, Email 等作为虚拟传感器, 而 Archive, Gopher, Netfind 等作为虚拟作用器, 这样就构成了 Internet 网络环境下的“软件机器人系统”。

所谓可移动智能代理 MIA (Mobile Intelligent Agent, 简称可移动代理) 是指在复杂的网络系统中能够从一台计算机移动到另一台计算机的程序, 该程序能够选择何时、何地移动; 在移动时, 该程序在仲裁点根据要求挂起其运行, 然后转移到另一台计算机上开始或继续运行, 最后把结果传回到代理的原用户。可移动代理系统中传输的对象并非是系统, 而是独立于运行平台的程序(可移动代理), 通过一定的传输机制(如 RPC, Email 等)把自己或其子代理传输到另一台计算机平台上, 采用这种方式传输的程序应相对于网络带宽较小。可移动代理是智能代理, 但它与一般智能代理的区别在于其可移动性。在设计移动代理系统时, 还应注意避免出现两个问题: 多代理对性能优良资源的集中争夺, 管理与协调因失效而引起系统崩溃或计算效率降低。我们认为可移动代理计算模型比传统的“客户机-服务器”模型具有如下优点:

(1) 可避免网络带宽浪费。因为可移动代理的部分远程工作是直接在服务器或远程计算机上运行, 并把最终结果传输给用户, 无需传输中间结果, 从而保留了网络的有限带宽, 这对于高延迟, 低带宽网络环境来说是十分有效的;

(2) 可充分利用服务器集合资源;

(3) 提高分布计算效率。由于它具有对环境的感知能力以及有效的传输机制, 从而可及时获取所需资源和实现最优计算方式;

(4) 容错性。由于移动代理无需计算机之间维护永久性的网络连接;

(5) 功能可扩充性。在移动计算模型中通过扩展移动代理功能, 并在服务器端运行, 则可在一定程度上体现服务器功能的扩充。

(6) 动态性(自适应性、自治性和协作性等)。可移动代理的自适应性表现在: 对任务求解需求实现的移动性、代理之间的协作性以及对环境感知能力; 可移动代理的自治性表现在代理可根据任务或事件驱动则可自行工作, 而无需人为干涉其启动和执行; 协作性是智能代理系统重要的特性, 可移动代理正是通过协作来实现其智能的增加和对复杂问题的求解;

(7) 代理的智能性。可移动代理的智能性表现在根据环境和任务需求规划行为, 以获取最佳问题求解效率。

设计可移动代理系统时主要采用的技术如下:

(1) 确保软件互操作的解释模块。在异种或异构平台上提供一层开放式代理执行环境, 把代理的代码解释(“映射”)为各平台可运行的代码集合;

(2) 传输基于各平台地址, 从而隐含了通道操作, 这有利于独立于通道存取; 另一方面, 地址为通信网络中基本操作元素, 如 IP 地址。对于代理来说主要关心的是操作平台地址, 而非传输方式;

(3) 代理流控制。如合理分配资源, 协调多代理间操作, 以防止多代理间严重冲突和死锁;

(4) 确保对代理的请求队列控制与管理。用户与代理间消息传递时, 必然会存在多请求消息情况, 因而必须有相应缓存管理, 如专门设计一个服务器或程序用于此项服务;

(5) 安全设计保证系统存取的可靠性, 系统的不安全因素包括非法代理存取资源, 恶意代理对资源(系统或其它代理)的破坏, 以及携带病毒代理对系统的影响等;

(6) 智能移动代理应能与其它代理和网络资源相互通信, 具有能获取周围环境知识和其内部状态的能力, 以进行自适应调整其行为等;

(7) 实现可移动代理与其它代理之间的协作机制。

2.1 可移动代理层次结构

可移动代理的抽象机模型见图 1 所示。移动代理采用代理编程语言如 Tcl, Java 等编写; 代理抽象规范机规定了数据操作类型和操作方法等, 这种规定是面向绝大多数平台支持的标准。在这个模型中, 代理抽象规范机实质上起着代理程序与在实际平台上运行程序的“映射”功能。在客户机端, 采用编译器检查可移动代理的静态错误, 对代理程序并不进行数值引用和内存分配等操作, 这是移动代理能够移动的关键; 在服务器端, 采用解释器实现可移动代理

程序向运行平台的“映射”。

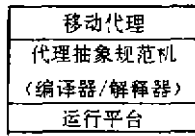


图 1 可移动代理抽象机

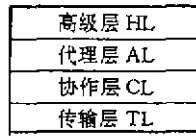


图 2 可移动代理系统层次结构

可移动代理系统可划分为四层:传输层、协作层、代理层和高级语言层(见图 2 所示)。传输层负责把代理由一机器传输到另一个机器;协作层实现代理之间协作控制;代理层实现代理的生成、管理以及代理之间的通讯对话等;高级层实现代理与用户之间交互。在此体系结构中,相邻的低层向上层提供无差错的透明服务。

2.2 可移动代理工作原理

可移动代理工作模式为:移动-运行和运行-移动。目前分布式信息检索系统中可移动代理主要采用前者,主要通过代理携带任务并由代理本身或其子代理移动到服务器环境下运行,然后把运行结果传送回用户。“运行-移动”工作模式为代理接受任务后则对任务进行求解,当继续执行条件不能满足(如所需资源本地没有等)时,则该代理分析现行网络环境找出所需资源所在机器,然后把自己或其子代理移动到目的机器上运行并把结果传回到原机器上,如此直到整个任务完成为止,再把结果返回给用户。

可移动代理工作状态主要包括:就绪,运行/协作,挂起,阻塞,移动和终止,其状态流程见图 3 所示。

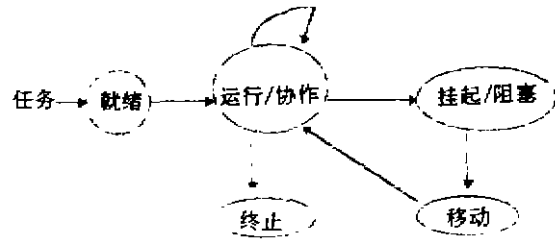


图 3 可移动代理状态迁移图

MIA 可采用四元组表示:

$MIA(S_m, D_m, Status, Abilities)$

其中: S_m 为源计算机集合, $sm_i \in S_m$; 对于客户机-服务器模式来说, sm_i 为客户机;

D_m 为目的计算机集合, $dm_i \in D_m$ 。对于客户机-服务器模式来说, dm_i 为服务器,它是 MIA 所移动的计算机集合;

Status 为 MIA 的内部状态集合,它可由 MIA 自己提取和分析。Status = {Block, Run, Suspend, Done, Move};

Abilities 为 MIA 的能力集合,它包括若干 MIA 的操作,如 Abilities = {MIArun, MIAgetstatus, MIAjump, ...}。

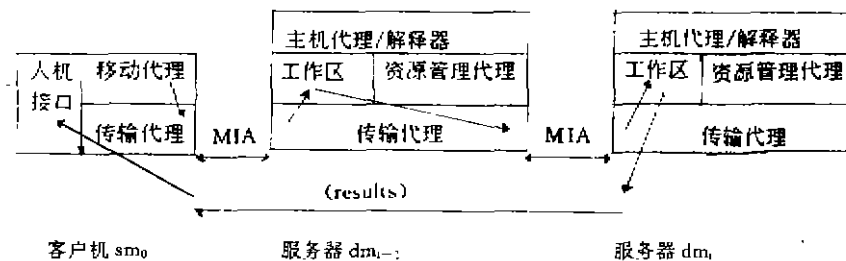


图 4 基于客户机-服务器的可移动代理工作模式

上述 MIA 系统工作基本原理(见图 4 所示)可描述如下:

```

GLOBAL MIA, Host.Agent.1, Host.Agent.2;
PROC MIA(event/task);
{...return $result};
AGENT.BEGIN
dmi-1 = sm0;
MIA.status = 'ready';
    
```

```

MIA_trigger(MIA, Event/Task);
REPEAT
MIA_run(MIA, event/task, dmi-1);
MIA_get_status(MIA-internal-status);
IF (MIA_get_status == block) AND (MIA_analysis(MIA-run on dm1) == true) THEN
{ Agent_create_child_agent ( MIA, SUB
(MIA));
    
```

```

MIA_submit (SUB (MIA), [ MIA_ip/Vars
    v1,v2,...vn/Proc MIA ],dmi-1,dmi);
dmi-1:=dmi;
MIA:=SUB(MIA);}ELSE
    IF (MIA_analysis (MIA-run on dmi) =
        =false)THEN
        { MIA_sent_result (MIA, error,
            dmi-1:sm0);
            MIA.status:=done;}
UNTIL (MIA_get_status(task) ==Done);
MIA_sent_result (MIA, result, dmi-1:
    sm0);
Agent_Cancel(SUB(MIA);dmi-1);
AGENT-END;

```

目前较为典型的可移动智能代理系统有由 Traoma 大学和 Cornell 大学研制的 Tacoma, 由 General Magic 公司研制的 Telescript, MO 和 IBM Itinerant Agent 等, 这些系统均为 94 年左右公布的, 目前这些系统仍在继续开发中。

Tacoma 的代理采用 Hcl/Horus 语言编写, 采用组通讯和“错技术”, 一个代理可采用“Meet”操作来执行另一个代理。该系统服务主要由各代理提供, 如用于服务的电子付帐、重启动丢失代理、调度和目录服务等。它主要存在的问题是: (1) 不支持执行代理的中断, 即移动代理在新机器上不是从移动点开始运行的而是从头开始执行, 因此每个代理必须保留其状态信息; (2) 没有安全机制; (3) 其 Horus 部件不能用于多平台, 就限制了其通用性。

Telescript 为一面向对象的智能代理语言系统, 移动功能是其基本操作。代理可采用“Go”指令实现移动。它已经用于 AT&T 的 PersonaLink 网络。Telescript 主要问题是安全机制, 目前它主要应用于电子商务。Telescript 是目前第一个商业移动代理语言系统。

MO 系统允许代码段(片)发送到远程机并运行, 每个机器提供的运行环境包括一个解释器、若干

同步原语和一个共享数据目录。MO 用于分布式系统的低级系统, 用于可移动代理系统最低层。

其它还有 Oracle 的移动代理, Safe-Tcl/MIME (把 Tcl 程序嵌入到 Email 中, HotJava 浏览器(把 Java 程序嵌入到 Web 文本中), Sodabot (动态地分布其部件), IBM 的 Itinerant Agents 把移动代理与基于知识资源检索相结合等, 其中有的系统无法提供仲裁移动和多平台运行, 有的系统不支持安全机理并且自适应能力(环境的感知能力与决策能力等)较差。

结束语 上面已经讨论了可移动代理的基本概念、工作原理以及层次结构, 这些是已有的可移动代理系统的共性。由于可移动代理系统及其技术正在研究之中, 仍存在很多不完善的地方, 从目前研究现状来看, 此方向研究将主要集中在四个方面: (1) 代理语言及其开发系统的研究与完善; (2) 代理性能与网络特性之间的关系研究, 以确保可移动代理的动态性和自适应性; (3) 移动仲裁机制研究; (4) 安全机制研究, 以防止恶意代理或系统侵害。

可移动代理技术已经应用于网络管理、分布式信息检索、 workflow 管理、PDA 等领域, 我们相信对其进一步研究有利于分布计算和网络工作模式的改革。

鸣谢 衷心感谢导师张仲俊院士生前的关心和精心指导。

主要参考文献

- [1] Kenneth E. Harker, TIAS: A Transportable Intelligent Agent System, Tech Report, Dartmouth College, June, 1995
- [2] Robert S. Gray, Intelligent Agent, A Technology and Business Application Analysis, Tech. Report, Dartmouth College, Nov. 30 1995
- [3] Michael Wooldridge et al., Intelligent Agent: Theory and Practice, Tech. Report, Manchester Metropolitan University, Jan 1995
- [4] 王杰、白英彩, 智能代理网络的网络模型, 《计算机工程》, 1996 年增刊