

基于多邮箱的移动 Agent 通信算法

贺敏伟^{1,2}, 扶卿妮³, 李贵海³, 林健²

(1. 广东商学院信息学院, 广州 510320; 2. 北京航空航天大学经济管理学院, 北京 100083; 3. 五邑大学信息学院, 江门 529020)

摘要: agent 移动的可靠性、实时性是移动 agent 通信中的难点。该文在总结已有算法的基础上, 引入多邮箱和主机 Controller 的概念, 结合两者实现了一个高效、可靠的移动 agent 通信算法, 能保证通信的可靠性和实时性, 使得移动 agent 的迁移变得简单且完全自由。

关键词: 移动 agent; 多邮箱; 通信机制; 通信失效; 实时性

Mobile Agent Communication Algorithm Based on Multi-mailbox

HE Min-wei^{1,2}, FU Qing-ni³, LI Gui-hai³, LIN Jian²

(1. School of Information, Guangdong University of Business Studies, Guangzhou 510320; 2. School of Economics & Management, Beijing University of Aeronautics & Astronautics, Beijing 100083; 3. School of Information, Wuyi University, Jiangmen 529020)

【Abstract】 Reliability and real-time capability are the difficulties in mobile agent communication. A high-efficient and reliable mobile agent communication algorithm is realized by introducing the concepts of multi-mailbox and host controller based on existing algorithms. The new algorithm guarantees communication reliability, real-time capability and makes the moving of mobile agent simple and free.

【Key words】 mobile agent; multi-mailbox; communication mechanism; communication failure; real-time characteristic

移动 agent 技术包括通信、迁移、安全^[1]等机制, 通信机制是其中的关键技术^[2]。通信的可靠性、实时性是移动 agent 通信研究的热点, 本文主要研究如何保证移动 agent 通信的可靠性, 并在一定程度上实现通信的实时性。

1 问题的提出

目前 agent 通信机制大致分为广播方式、基于 Home 的消息转发机制、路径转发机制、基于邮箱的通信机制等。

广播方式将网络看成以消息发送者为起点向相邻节点发送信息, 相邻节点以同样的方式向各自相邻节点发送信息, 最终到达目标节点。可是当目标 agent 正在迁移, 同时消息正沿着目标 agent 迁移的反方向传递, 消息将不会被传递到 agent。

基于 Home 的消息转发方式是在 agent 迁移过程中, 发往 agent 的异步消息都保存在 Home 中, agent 向 Home 汇报它的新通信地址后, Home 再把消息发送给 agent。Home 转发方式使 Home 成为消息的聚集地。Home 主机创建的所有 agent 的异步消息由 Home 保存。由于对 Home 过分依赖, 很有可能使 Home 成为瓶颈。

在按路径转发方式中, agent 途径的每一个节点都记录了 agent 下一步迁移的地址, 消息沿着 agent 迁移的轨迹发送给 agent。但当迁移路径很长时, 消息发送的时延会增大, 存储迁移地址开销也会增大, 而且会出现消息永久追踪现象。

基于邮箱的通信方式中引入了“邮箱”(Mailbox)的概念。通常, 邮箱被定义为独立于移动 agent 的一个消息代理, 它用于暂时存储发给 agent 的异步消息。邮箱可以和 agent 分处不同主机。

由于 agent 在迁移时还有邮箱作为它的接收地址, 使得这种机制较为灵活。但是各种算法思想的不同以及对邮箱功能规定等方面的差异, 使得此类算法各有利弊。

文献[3]提出了基于邮箱的通信算法, 将邮箱状态分为迁

移态和静止态 2 种。邮箱处于静止态时, 可以接收发给 agent 的异步消息。邮箱处于迁移态时, 表示它迁往 agent 所在的主机, 消息将会阻塞。算法中邮箱的迁移次数少于 agent 的迁移次数, 增大了通信效率。缺点是当邮箱迁移时, 发往 agent 的消息会被阻塞; 当邮箱所在节点发生故障时, 所有保存在邮箱中的消息都将丢失; 实时性也不高。该算法的改进算法规定邮箱不随移动 agent 一起迁移, 并且当邮箱中的消息都转发给移动 agent 之后, agent 销毁邮箱, 避免了邮箱迁移时的消息阻塞。但每次是否摧毁邮箱都要经过 agent 向邮箱所在的主机发送查询消息才能决定, 邮箱所在的主机没有自动查询邮箱信息的功能。agent 迁移时, 向 Home 主机发送消息, 把通信地址修改为邮箱所在主机地址, 同时通知邮箱停止发送消息, 接着迁移到目标主机。如果在 agent 向邮箱发送停止消息的同时, 邮箱恰好向 agent 发送消息, 该消息会因为 agent 的迁移而丢失。发送者 agent 也可能存在同样的情况。因此, 该算法的致命缺点是其迁移不受限可能会导致通信失效^[4]。

针对上述算法的优缺点, 本文提出了一个迁移完全不受限、通信可靠的多邮箱通信算法。

2 改进的移动 agent 通信算法

2.1 基本前提

目前针对移动 agent 通信机制的研究大部分以不考虑网络传输故障为前提, 本文也假设在物理网络上传输的数据总能在有限时间内从信道的源端到达目的端, 并且这个信道是

基金项目: 国家博士后基金资助项目(2005038313); 广东省自然科学基金资助项目(010475); 广东省教育厅自然科学基金资助项目(Z202064)

作者简介: 贺敏伟(1963-), 男, 教授、博士后, 主研方向: 计算机网络, 复杂系统仿真; 扶卿妮、李贵海, 硕士研究生; 林健, 教授、博士、博士生导师

收稿日期: 2008-02-20 **E-mail:** heminwei@163.com

先进先出的。

2.2 数据结构

命名机制采用Mogent^[5]系统中的逻辑名和物理名层空间的命名机制。在该机制下，一个Mogent同时拥有逻辑名和物理名：逻辑名由生成移动agent的主机的IP地址或域名和用户赋予的一个Mogent名构成，实现了命名的全局唯一性。物理名由agent当前所在主机的IP或移动agentID构成，其中，移动agentID是该主机上的移动agentServer为移动agent所分配的唯一标识。这样，移动agent每迁移到一台主机，物理名都会改变，但是逻辑名在其整个生存期中保持不变。

将消息分为非常消息和普通消息，如果发送者 agent 发送的是非常消息，则在发送消息的同时用 Y 标记该消息，如果是普通消息，则不做此处理。

本系统包括移动 agent, Home, Controller 及 MAILBOXS 4 种实体。

(1)移动 agent 有一个自带邮箱记录，用以记录它所拥有的邮箱的主机地址。

(2)Home 记录本地节点创建的 agent 的动态信息，主要是 agent 的通信地址，每个 agent 的通信地址都保存在它出生节点的 Home 中。

(3)MAILBOXS 是一个 agent 的智能邮箱群，它包含 Mailbox(1),Mailbox(2) ... ,Mailbox(n) ，其中， $n \geq 1$ 。Mailbox(i)($i \in [1,n]$)是该 agent 的一个通信邮箱，它保存了 agent 从该节点离开之后到到达另一节点并向 Home 成功注册新地址之前所有发往该 agent 的消息。规定，一个 agent 在一个节点最多只能有一个邮箱。邮箱用一个消息队列来暂时保存发给 agent 的信息，消息队列如图 1 所示，其中，带 Y 标记的非常消息排在队列靠前的位置，普通消息排在队列靠后的位置。由 Controller 决定到达的消息在队列中的具体位置。

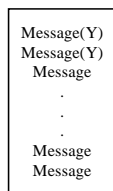


图 1 消息队列

(4)Controller 是一个具有一定智能的控制器，其结构如图 2 所示。

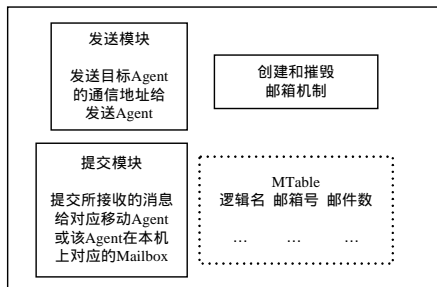


图 2 Controller 的基本结构

Controller 有 3 个功能：

- (1)负责通信，把目标 agent 的通信地址告诉发送方 agent。
- (2)负责将所接收的本节点上运行的移动 agent 消息提取到对应的移动 agent 或该 agent 对应的邮箱。如果消息是提取到邮箱，Controller 还控制消息在邮箱消息队列中的位置，将

带 Y 标志的消息插入到队列中第 1 条普通消息的前面，如图 3 所示。如果是普通消息，则直接插入到队列末尾，如图 4 所示。

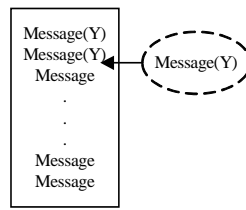


图 3 非常消息的插入

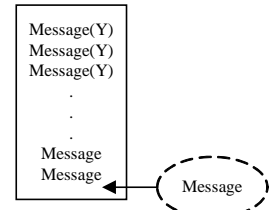


图 4 普通消息的插入

(3)创建和摧毁邮箱。在 Cotrollor 中创建了一个邮箱信息表(MTable)，用于记录本机上现存所有邮箱的邮箱状况和与这些邮箱对应的 agent 的逻辑名。

一个 agent 到达一个新的主机时，会向它的 Home 注册新的通信地址。当一个移动 agent 离开本主机时，本主机的 Cotroller 根据该 agent 的逻辑名查询 MTable。如果没有查询到与该逻辑名相关的记录，表示该 agent 在该主机没有邮箱，Cotroller 会创建一个与之对应的邮箱，用于暂时保存发给该 agent 的消息；如果能查询到，则表示该主机上有该 agent 的邮箱，此时，用查询到的邮箱号对应的邮箱继续保存发给该 agent 的消息。

在邮箱信息表中除了保存邮箱号和 agent 的逻辑名外，还保存了与邮箱号对应的邮箱消息队列中的邮件数，它是 Controller 决定是否销毁邮箱的依据。

2.3 算法

注册过程如下：

步骤 1 移动 agent 到达一个新节点后，会向 Home 主机注册新的通信地址 ADR，若自带邮箱记录不为空，向自带邮箱记录中的所有主机发送“通知”，通知该 agent 对应的邮箱开始发送消息。“通知”中包含有本机的逻辑地址和 agent 的逻辑名。

步骤 2 Home 主机收到新的通信地址后，把该移动 agent 的通信地址 ADR 修改为新地址。

若移动 agent 自带邮箱记录不为空，则其中记录的主机收到“发送”通知后，主机的 Controller 查询 MTable：如果 MTable 中有逻辑名与“通知”中所带逻辑名相符的记录，则表示存在该 agent 对应的邮箱；否则表示已不存在。

若不存在对应的邮箱，则给该 agent 发送一个“死亡”通知。若存在，Controller 判断邮箱是否为空：(1)当邮箱不为空，即邮箱信息表中邮箱对应邮件数不为 0 时，向 agent 发送消息，每发送一条消息，邮箱信息表中该邮箱对应的邮件数减 1；当邮箱中邮件数变为 0 时，转(3)。(2)若邮箱为空，即邮箱信息表中邮箱对应邮件数为 0 时，转到(3)。(3)等待一个 T 时间后，再查询邮件数是否为 0，若此时邮件数不为 0，转(1)；若邮件数仍然为 0，则“消灭”此邮箱，并向 agent 发送“死亡”通知。

步骤 3 如果 agent 收到“死亡”通知，清除 agent 自带邮箱记录中关于此邮箱的记录。

通信过程如下：

步骤 1 发送 agent 向目标 agent R 的 Home 主机寻址。

步骤 2 Home 主机返回目标 agent R 的通信地址 ADR。

步骤 3 发送 agent 把消息发向地址为 ADR 的 agent R。

步骤 4 通信地址 ADR 所在的主机 Controller 的提交模块对收到的消息进行判断：如果移动 agent 在本主机上，直接提交给它；如果不在本主机上，则提交给它的邮箱。提交给邮箱时也要进行判断：如果消息是普通消息，直接插入到邮箱消息队列的末尾(如图 4 所示)，邮箱信息表 MTable 中该邮箱的邮件数加 1；如果是带 Y 标记的非常消息，则插入到第 1 个普通消息之前(如图 3 所示)，邮箱信息表 MTable 中该邮箱的邮件数加 1。

迁移过程如下：

步骤 1 移动 agent 通知其所有邮箱停止发送消息。

步骤 2 agent 所在主机的 Controller 进行判断：若邮箱信息表中有该 agent 的记录，则转(3)；如果邮箱信息表中没有该 agent 的记录，则为其创建一个邮箱，在邮箱信息表中添加一条新记录，记录该 agent 的逻辑名和其在本机上拥有邮箱的邮箱号，并把该机地址保存在移动 agent 的自动邮箱记录中。

步骤 3 移动 agent 迁移到目标主机。

假设移动 agent R 从主机 C 迁移到主机 D；agent R 的自带邮箱记录如下：

主机地址
IP<Host B>
IP<Host C>

主机 B 的邮箱信息表 MTable 的情况如下：

逻辑名	邮箱号	邮件数
IP<Host A>+R	1	1
IP<Host C>+S	2	3
...

主机 C 的邮箱信息表 MTable 的情况如下：

逻辑名	邮箱号	邮件数
IP<Host A>+M	1	5
IP<Host B>+T	2	1
IP<Host A>+R	3	2
...

这时 agent 的注册流程如图 5 所示，其中，移动 agent R 的逻辑名为：IP(Home<R>)+R。

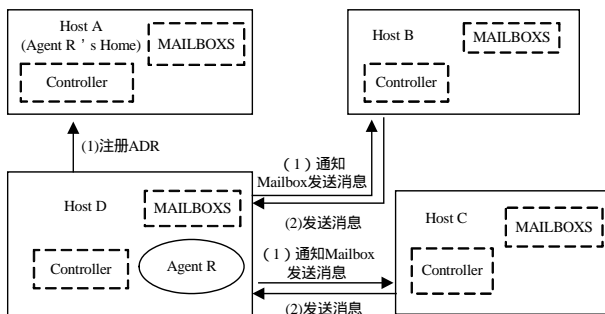


图 5 agent 的注册流程

迁移过程和通信过程与之类似，这里不再赘述。

2.4 算法分析

算法具有如下特点：

(1)每个节点有一个 Controller，封装了许多 agent 迁移和通信时的行为以及邮箱管理功能，如创建和消灭邮箱，使得 agent 之间的通信变得简便而可靠。

(2)移动 agent 在迁移时，只须通知邮箱停止发送消息和确保本地主机有其对应的邮箱即可迁移，其他动作都交给本机 Controller，使得迁移非常方便；因为不需要向 Home 发送迁移请求和地址注册，所以减少了地址更新的次数及移动 agent 迁移的开销。

(3)移动 agent 迁移后，因为在迁移前的主机上留下了邮箱，发送者 agent 发送过来的消息不会因为 agent 迁移而丢失，而且移动 agent 通知所有邮箱停止发送消息的同时邮箱向 agent 发送的消息以及之前发送的还未到达的消息都会被邮箱接收，不会因为 agent 的迁移而丢失。

(4)空邮箱能被高效、及时地处理。当邮箱中的消息发送完后，不需要移动 agent 的参与，邮箱所在本机上的 Controller 能自动侦测并“消灭”空邮箱。因为动作在本机上进行，消灭空邮箱的速度很快，避免了因为网络消息丢失等问题而可能产生的“消灭”不成功的现象。

(5)具有一定的实时性。将消息分为非常消息和普通消息，让非常消息插入到信息队列的前面，使得 agent 首先接收到非常消息，便于及时处理非常消息。

(6)引入了多邮箱的概念，虽然短时间内会同时存在几个不等的邮箱，占用一定的主机资源，但是一段时间后，消息发送完的邮箱会“死亡”，剩余的邮箱会很少甚至为 0。而且因为消息分布在多个邮箱即多个节点上，所以当某个节点出现问题甚至导致崩溃时，信息丢失得比较少，鲁棒性比单邮箱机制好。

(7)为了让还未到达邮箱的消息不丢失，设置了一个 T 时间，其取值一般为 1 个~2 个消息的平均传输时间。

3 结束语

本文提出了一种多邮箱通信算法，在保证 agent 迁移自由的条件下，避免了通信失效现象，通过设置消息队列使系统具有一定的实时性，能在一定程度上保证非常消息的及时传递。但其在寻找 agent 逻辑名时对 agent 的 Home 有很强的依赖性，对实时性的研究也不太深入，这两方面将是下一步的研究工作。

参考文献

- [1] 杨 鲲, 刘大有, 郭 欣. 一个具有高安全性的移动 Agent 系统模板结构[J]. 软件学报, 2002, 13(1): 130-135.
- [2] 唐浩坤, 汪林林. 移动 Agent 可靠位置透明通信方法的研究[J]. 计算机工程, 2006, 32(15): 126-128.
- [3] Feng Xinyu, Cao Jiannong, Lu Jian, et al. An Efficient Mailbox-based Algorithm for Message Delivery in Mobile Agent Systems[C]//Proceedings of the MA'01. Berlin, Germany: Springer-Verlag, 2001.
- [4] 周竞扬, 陈韬略, 许 平, 等. 移动 Agent 系统通信失效问题研究[J]. 计算机科学, 2003, 30(8): 108-111.
- [5] 陶先平, 冯新宇, 李 新, 等. Mogent 系统的通信机制[J]. 软件学报, 2000, 11(8): 1060-1065.