

# 面向 LEO 卫星网的轻量级按需 QoS 源路由算法

朱 军<sup>1</sup> 饶 元<sup>1,2</sup> 李绍稳<sup>1</sup> 邵 星<sup>2</sup>

(安徽农业大学信息与计算机学院 合肥 230036)<sup>1</sup> (南京邮电大学计算机学院 南京 210003)<sup>2</sup>

**摘 要** 接入卫星切换、流量分布不均匀和星上处理能力受限等因素使 LEO 卫星网的 QoS 路由面临巨大挑战。提出了一种引入移动代理技术的轻量级按需 QoS 源路由算法。首先结合星际链路队列占用情况设计了星际链路可用指数;接着基于卫星网承载流量分布特点提出了卫星可用指数;综合星际链路和卫星可用指数给出了移动代理迁移策略、星际 QoS 路由及其重建算法。仿真结果表明,提出的路由算法信令开销低,时延抖动、新呼叫阻塞率和切换呼叫阻塞率等 QoS 性能明显优于传统路由算法。

**关键词** LEO 卫星网,移动代理,QoS,路由算法

**中图分类号** TP391 **文献标识码** A

## Lightweight On-demand QoS Source Routing for LEO Satellite Network

ZHU Jun<sup>1</sup> RAO Yuan<sup>1,2</sup> LI Shao-wen<sup>1</sup> SHAO Xing<sup>2</sup>

(College of Information and Computer, Anhui Agricultural University, Hefei 230036, China)<sup>1</sup>

(School of Computer Science & Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)<sup>2</sup>

**Abstract** QoS routing for LEO(Low Earth Orbit) satellite network faces great challenges due to several issues, such as, ingress satellite handover, unbalanced distribution of traffic and the limited on-board processing ability. A lightweight on-demand QoS source routing algorithm using mobile agent was proposed. Firstly, ISL(inter-satellite link) availability index was designed considering ISL utility. Secondly, satellite availability index was put forward based on traffic distribution undertaken by satellite network. Thirdly, the migration strategy of mobile agent was given taking both indexes into account. Consequently, ISL routing and re-routing were presented in detail. Simulation results demonstrate that the proposed approach has lower signaling overhead, better delay jitter, new call blocking probability and handover blocking probability than the traditional methods.

**Keywords** LEO satellite network, Mobile agent, QoS, Routing algorithm

## 1 引言

低轨 LEO(Low Earth Orbit)卫星网络中各卫星节点处于高速运动状态,这使其 QoS 路由面临巨大的挑战,具体表现在:接入卫星切换造成网络拓扑频繁变化、地球表面人口分布的不均匀造成卫星承载的流量非均匀分布、星际链路高传播时延造成链路状态实时性不够<sup>[1]</sup>。唐剑等人提出由中轨道卫星网承载部分 LEO 卫星流量的 QoS 路由<sup>[2-4]</sup>,但卫星之间可视角变化频繁、天线对准和跟踪技术的限制致使层间星际链路不稳定,多层 QoS 路由在具体实现时有很大的困难。因此研究单层 LEO 卫星网的 QoS 路由更有意义,近年来这方面的研究已经取得了一些成果。

概率星际路由<sup>[5]</sup>采用星际链路传播时延作为路径度量,使用最短路径算法计算路由,其忽视了网络流量的变化,当负载较重时链路拥塞使得找到的路径往往不可用。洪泛源路由机制<sup>[6]</sup>能够得到较多的候选路径,从而具有较高的 QoS 路径

探测性能,但其向网络广播信令,使得开销巨大。为此, Xu Hui 等人<sup>[7]</sup>提出了适用于 ATM 卫星网的分布式蚂蚁 QoS 路由,其一定程度上缓解了信令开销,但缺乏切换路由重建机制。Karapantazis S. 等人<sup>[8]</sup>结合 Iridium 系统特点修改距离矢量路由协议 AODV 的路由更新策略,提出了按需多服务路由 MOR;但 MOR 需要大量的存储空间来缓存等待转发的数据包。卫星网多协议标记交换路由<sup>[9]</sup>性能较高,但未能解决如何降低网络拓扑频繁变化引起的路由重建和标签交换路径维护开销的问题。鉴于卫星网拓扑和流量分布的快速变化, Huang Q 等人<sup>[10]</sup>认识到合适的流量分配机制的缺乏是端到端时延和带宽等 QoS 性能无法保证的根本原因,因此提出了高性能卫星路由 HPSR(High Performance Satellite Routing),其一次呼叫接入在全网范围内重新分配带宽、计算路由,获得了较高的吞吐量,但计算、信令开销过大。

基于星上资源受限性,卫星网 QoS 路由算法应具有简洁有效、低计算复杂度、低信令开销、低存储空间和收敛迅速等

到稿日期:2011-08-25 返修日期:2011-11-09 本文受国家自然科学基金(60973139,30971691),安徽省高校省级自然科学基金研究重点项目(KJ2012A102)资助。

朱 军(1972-),男,博士生,讲师,主要研究方向为自组网路由、数字农业,E-mail:zhujun@ahau.edu.cn;饶 元 博士后,讲师;李绍稳 教授,博士生导师;邵 星 博士生。

特点。此外,接入卫星切换增加了时延和通信中断的概率。从用户的角度来看,通信中断比新呼叫的拒绝更难以忍受,卫星网 QoS 路由算法设计过程中不仅要考虑系统的新呼叫接入能力,还需考虑切换成功率。现有的 LEO 卫星网 QoS 路由算法不同程度地存在计算、信令开销过大, QoS 指标考虑不足的问题,尚处于初级探索阶段。

本文引入移动代理技术,从 LEO 卫星星座特点及流量分布情况入手,研究时延和带宽双约束条件下具有低开销、低时延抖动、低呼叫阻塞率和切换阻塞率的轻量级 QoS 路由算法。

## 2 相关定义

本文以 Walker 星座为研究对象,星际链路始终可用,将卫星网抽象为一个无向连通图  $G=(V,E)$ ,其中  $V$  表示卫星网络中所有卫星节点集合, $E$  表示星际链路 ISL(Inter-Satellite Link)集合。对于任一卫星  $u$ ,其邻居卫星集合  $N(u)$  可以表示为  $N(u)=\{v|ISL_{u,v}\in E,u,v\in V\wedge u\neq v\}$ ,其中  $ISL_{u,v}$  为相邻卫星  $u$  和  $v$  间的星际链路。

输出缓冲队列使用情况最能反映卫星节点承载的流量负载变化。由于流量负载变化具有突发性和连续性,因此本文采用星际链路队列的统计占用量来描述流量负载的变化。设  $q_{utility_{u,v}}(t)$  表示  $t$  时刻卫星  $u$  与某一邻居卫星  $v$  间的星际链路  $ISL_{u,v}$  队列的统计占用量,则:

$$q_{utility_{u,v}}(t)=\frac{1}{\Delta t}\int_{t-\Delta t}^t q_{u,v}(i)\times di \quad (1)$$

式中, $q_{u,v}(i)$  表示  $i$  时刻星际链路  $ISL_{u,v}$  队列的占用量。

考虑到时延参数可加性,则卫星  $u$  和卫星  $v$  之间的  $ISL_{u,v}$  在  $t$  时刻的复合链路时延  $ISLDelay_{u,v}(t)$  可以表示为传播时延和队列时延之和,即

$$ISLDelay_{u,v}(t)=PD_{u,v}(t)+QD_{u,v}(t) \quad (2)$$

式中, $QD_{u,v}(t)$  表示星际链路传输时延,且  $QD_{u,v}(t)=\frac{q_{utility_{u,v}}(t)\times P_{avg}}{C}$ ,  $C$  为星际链路带宽,  $P_{avg}$  为数据包平均大小。传播时延  $PD_{u,v}(t)$  用来描述星际链路长度变化,表征网络中不同轨道面上卫星之间的互连关系,可由卫星网拓扑状态提前计算得到。

**定义 1**(星际链路可用指数, ISL Availability Index, IAI)

设  $q_{capability_{u,v}}$  表示星际链路  $ISL_{u,v}$  队列的总长度,则该星际链路的可用指数  $IAI_{u,v}(t)$  为:

$$IAI_{u,v}(t)=\begin{cases} 2\times q_{capability_{u,v}}, & q_{utility_{u,v}}(t)=0 \\ \frac{q_{capability_{u,v}}}{q_{utility_{u,v}}(t)}, & \text{其他} \end{cases} \quad (3)$$

由定义 1 可知,星际链路可用指数基于星际链路使用比率来定义,可用指数越大,表示该星际链路越空闲,相应的链路可用性也越大。

鉴于卫星承载的流量与地理位置密切相关,本文在文献 [11] 所提出的热点区域模型基础上,提出了融合卫星所处地理位置的卫星可用指数。

**定义 2**(卫星可用指数, Satellite Availability Index, SAI)

令  $lat_v$  表示时刻  $t$  卫星  $v$  所处的纬度,则卫星可用指数  $SAI_v(t)$  为:

$$SAI_v(t)=\begin{cases} e^{lat_v/90}, & -90\leq lat_v < lat_c \\ e^{-(lat_v-2\times lat_c)/90}, & lat_c\leq lat_v\leq 90 \end{cases} \quad (4)$$

式中, $lat_c$  表示北半球热点区域的中心纬度。

综合星际链路可用指数和卫星可用指数,本文定义了移动代理迁移过程中选择邻居卫星作为下一跳的概率计算方法。

**定义 3**(卫星选择概率, Satellite Selection Probability for next hop, SSP) 令  $SSP_{u,v}(t)$  表示移动代理在卫星  $u$  上选择邻居卫星  $v$  作为下一跳的概率,则:

$$SSP_{u,v}(t)=\alpha\times\frac{IAI_{u,v}(t)}{\sum_{i\in N(u)}IAI_{i,v}(t)}+\beta\times\frac{SAI_v(t)}{\sum_{i\in N(s)}SAI_i(t)} \quad (5)$$

式中, $\alpha,\beta\in[0,1],\alpha+\beta=1$ 。定义 3 首先使用式(3)、式(4)计算各星际链路和卫星的可用指数,然后使用  $\alpha$  和  $\beta$  来调节两种星际链路指数,使用两者加权和来计算选择邻居卫星并将其作为下一跳的概率。

## 3 轻量级按需 QoS 源路由算法

轻量级按需 QoS 源路由算法 LQSR(Lightweight on-demand QoS Source Routing)选择具有最大覆盖时间的卫星作为源、目的接入卫星。一旦选定接入卫星,LQSR 就启动星际 QoS 路由算法建立源、目的卫星间的数据传输路径;通信期间接入卫星发生切换时,LQSR 启动路由重建算法来重新建立数据传输路径。

### 3.1 星际 QoS 路由算法

星际 QoS 路由过程中,源卫星节点首先派遣移动代理在网络中迁移探测多条可行的候选星际 QoS 路径,然后从候选星际路径集中选择一条路径用于数据传输。星际 QoS 路由使用两种移动代理:前向 LQSR 代理和后向 LQSR 代理。源接入卫星派遣指定寿命的前向 LQSR 代理向目的接入卫星迁移探测满足 QoS 指标的路径。一旦到达目的接入卫星,前向 LQSR 代理就自动转换成后向代理。后向代理往返迁移并沿途预留所需带宽。

星际 QoS 路由算法流程如图 1 所示,详细描述如下:

(1) 源卫星节点  $s$  按一定的时间间隔连续派遣  $k$  个指定寿命的前向 LQSR 移动代理  $F-QoS_{s\rightarrow d}$ ,这些移动代理在卫星网络中向目的接入卫星  $d$  迁移。

(2) 前向 LQSR 代理使用高优先级队列,以便快速探测 QoS 路径。在各卫星节点  $v$  上,LQSR 代理按照如下规则选择下一跳。

如果达到最大寿命,则前向代理自动销毁。

如果所有的邻居节点都已经访问,则前向代理自动销毁。

如果没有星际链路可以满足带宽要求,则前向代理自动销毁。

从所有可达且可满足带宽要求的星际链路、但尚未访问的邻居节点中,选择具有最大卫星选择概率 SSP 的卫星作为下一跳。

(3) 在向目的节点迁移的过程中,前向代理记录探测到的 QoS 路径情况。在每个被访问节点  $v$  上,前向代理获取连接当前节点和下一跳的星际链路时延(ISL Delay),并将其压入内存堆栈,然后使用式(6)计算路由总时延  $TotalDelay$ ,如果  $TotalDelay>D$ ,则前向 LQSR 代理自动销毁。

$$TotalDelay = ISLDelay_{s,v_1} + \sum_{i=1}^{m-1} ISLDelay_{v_i,v_{i+1}} \quad (6)$$

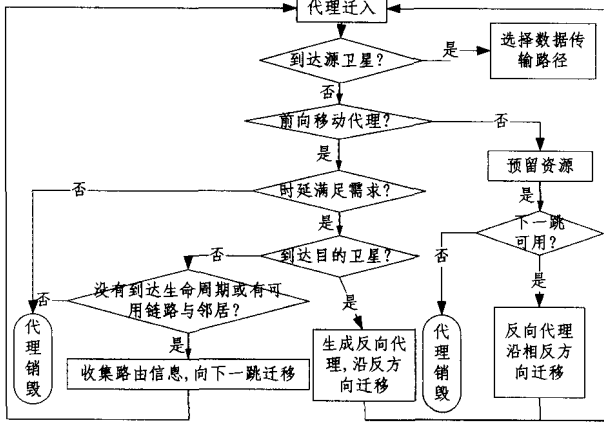


图1 星际 QoS 路由流程图

(4) 当前向代理到达目的节点时,其内存堆栈包括如下形式的 QoS 路由信息:  $\{(v_1, ISLDelay_{s,v_1}), (v_2, ISLDelay_{v_1,v_2}), \dots, (v_k, ISLDelay_{v_k,d})\}$ 。如果  $TotalDelay > D$ , 则前向 LQSR 代理自动销毁, 否则其自动转换成后向代理  $B-QoS_{s \rightarrow d}$ ; 后向移动代理沿着前向代理的迁移路径反方向迁移。

(5) 后向移动代理同样使用高优先级队列, 以便及时传递探测到的 QoS 路径。在各卫星节点, 后向代理弹出自身堆栈来获取下一跳, 通知卫星节点预留带宽并启动资源预留开关(超时仍然没有数据传输就释放预留的资源)。

(6) 源节点等待一定时间后, 如果还没有后向代理返回, 则拒绝呼叫请求。如果有多个后向代理返回, 则将各路由的总时延  $TotalDelay$  归一化处理后, 使用轮盘赌策略选择一条路径传输数据。

### 3.2 路由重建算法

LEO 卫星网中, 接入卫星切换不可避免, 这会导致正在传输数据的 QoS 路径失效。要使服务得以继续, 就必须进行路由重建, 这需要过渡时间, 增加了计算和信令开销。因此, 路由重建策略的设计必须综合考虑两方面因素: 一是必须减少路由重建过程对已中断业务性能的影响, 即保证最小切换等待时间, 以减少对中断业务 QoS 性能指标的影响; 二是必须减少路由重建对其它未中断业务的影响, 即路由重建不能消耗太多的星际链路资源和信令开销。

目前国内外学者提出的切换路由重建算法主要有 3 类: 路径截短、部分路径扩展和完全路径重建。若切换目标卫星在原路径中间, 则可采用路由截短策略。这种策略切换开销较小、处理时间短, 且不会造成业务 QoS 指标的下降和星际链路资源的浪费, 是最有效的切换策略。若切换目标卫星不在原路径中间节点, 则可以采取部分路径扩展或完全路径重建方案。路径扩展通过为中断路径建立延长路由的办法来保证已有路径的连续性, 使原有路径被部分保存, 只有变化部分被更新。该方案实现简单, 同时也减小了切换处理时间, 对中断路径性能影响较小。但新生成的路径可能不是最优路径, 这会造成对网络资源的不合理使用, 而且原有呼叫的时延指标在新路径中可能会被破坏。完全路径重建方案是用新路径完全替代原有路径, 优点是新路径仍然是最优化路径, 其能够

最合理使用网络资源; 缺点在于新路径的建立时延和信令开销较大。

切换目标卫星一般是切换前卫星的邻居或者离切换卫星跳数较近。本文优先进行路径截短, 再考虑部分路径的扩展, 最后才启动完全路径重建。源接入卫星、目的接入卫星都可能发生切换, 两种切换的路由重建步骤类似。下文以源卫星切换为例阐述路由重建步骤:

(1) 选择覆盖仰角满足要求且位于原先 QoS 路径中的卫星建立候选切换目标卫星集合。若该集合非空, 则选择剩余覆盖时间最长的卫星(如  $a_1$ )作为切换目标卫星, 新 QoS 路径为原先 QoS 路径中从  $a_1$  到  $d$  之间的部分:  $a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow d$ (见图 2 中 1 号截短路径)。否则, 执行(2)。

(2) 选择剩余覆盖时间最长的卫星作为切换目标卫星  $h_s$  (Handover Source satellite)。如果  $h_s$  是切换前卫星  $s$  的邻居节点且带宽和时延满足要求, 则新 QoS 路径为:  $h_s \rightarrow s \rightarrow a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow d$ (见图 2 中 2 号扩展路径), 否则, 执行(3)。

(3)  $h_s$  启动星际 QoS 路由算法, 建立一条从  $h_s$  到  $s$  的 QoS 路径, 此时新 QoS 路径为:  $h_s \rightarrow \dots \rightarrow s \rightarrow a_1 \rightarrow a_2 \rightarrow a_3 \rightarrow a_4 \rightarrow d$ 。如果探测失败, 则执行(4)。

(4)  $h_s$  使用星际 QoS 路由算法, 建立一条完整的从  $h_s$  到达  $d$  的新 QoS 路径(见图 2 中 3 号重建路径)。

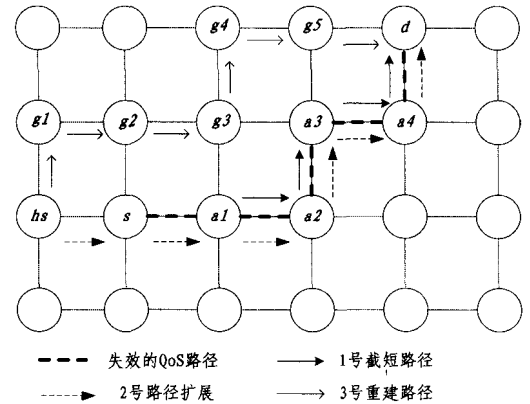


图2 星地链路切换路由重建

## 4 仿真实验与分析

### 4.1 参数设置

本节借助于 NS2 仿真工具<sup>[12]</sup>评价 HPSR<sup>[7]</sup>和 LQSR 算法性能,  $\alpha$  取值为 0.7,  $\beta$  取值为 0.3。地面终端数量为 300 个, 这些终端按照人口密度分布在全球, 并按照文献<sup>[11]</sup>给出的流量矩阵产生呼叫, 通过改变新呼叫到达率调整注入到网络中的流量负载。选择类 Courier 作为仿真星座, 参数设置如表 1 所列。对于新呼叫, 在呼叫等待时长内没有 QoS 路径返回, 就阻塞该呼叫; 对于呼叫切换, 在切换等待时长内没有 QoS 路径返回, 就阻塞该切换。

为客观对比算法, 本文按照路径截短优先、部分路径扩展次之、完全路径重建最后的原则完善了 HPSR 算法的路由重建模块。仿真中, HPSR 和 LQSR 的接入卫星选择与路径重建方法相同, 但星际 QoS 路由过程不同。HPSR 采用 Wang 算法(类似于最短路径优先算法)计算星际路径, LQSR 算法

派遣移动代理进行星际 QoS 路由。

表 1 参数设置

参数	取值
轨道数	8
卫星总数	72
轨道高度	1100km
最小仰角	10°
数据包大小	500Bytes
端到端时延	200ms
“on”周期	0.4s
“off”周期	0.8s
“on”周期内比特率	64kB/s
平均通信时长	5 min
呼叫等待时长	60s
切换等待时长	20s
前向 LQSR 代理派遣间隔	0.3s
每连接派遣 LQSR 代理数量	5 个

通过比较以下几个参数,分析路由算法的 QoS 性能。

(1) 归一化信令开销:路由代理(数据包)数量与终端产生的数据包之比。该指标反映了路由机制在不同呼叫强度下引入到网络中的信令开销。

(2) 呼叫阻塞率:被阻塞呼叫数(未能成功建立呼叫次数)与总呼叫次数之比。此指标直接反映路由机制的呼叫接入能力。

(3) 切换阻塞率:掉话次数(未能成功切换次数)与总切换次数之比。此指标直观地反映路由机制的流量平衡能力。

(4) 时延抖动:连续到达目的节点的数据包间的时延差。此指标可用来评价算法提供路径的稳定性。

## 4.2 算法 QoS 性能分析

### (1) 归一化信令开销

图 3 给出了不同新呼叫强度下的归一化信令开销变化情况。LQSR 算法明显优于 HPSR,一次 QoS 业务新呼叫,LQSR 算法仅派遣 5 个移动代理探测 QoS 路径;而 HPSR 算法则要收集全网星际链路状态,即使采用类似 ELB<sup>[13]</sup>的链路状态分发机制,开销仍然巨大。

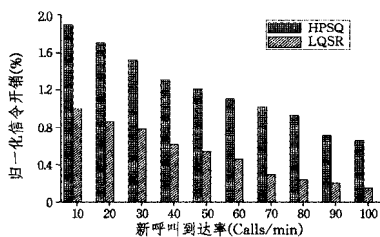


图 3 不同呼叫强度下归一化信令开销

### (2) 新呼叫阻塞率

图 4 分别给出了不同呼叫强度下 2 种算法的呼叫阻塞率 CBP(Call Blocking Probability)。呼叫到达率低于 10calls/min 时,2 种算法阻塞率基本一致。随着呼叫强度的增加,LQSR 算法的新呼叫阻塞率低于 HPSR。这是由 HPSR 的 QoS 路由特点所决定的:当呼叫到达时,HPSR 算法首先分配具有最小带宽且端到端时延低于业务需求的路径,若分配的最小带宽低于业务请求带宽,则在全网范围内重新分配所有的连接带宽。呼叫强度低时,首次分配成功率较高;随着呼叫强度的增加,首次分配成功率逐渐下降,全网范围内重新分配带宽次数增多,而全网带宽分配时间往往超过呼叫等待时间,

导致新呼叫被阻塞。

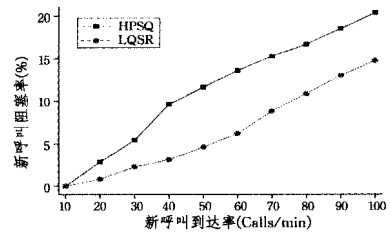


图 4 不同呼叫强度下呼叫阻塞率

### (3) 切换阻塞率

仿真结果表明,切换阻塞率 HBP(Handover Blocking Probability)都低于呼叫阻塞率 CBP(见图 5),这是因为发生星地链路切换时,新的源(或目的)卫星可能是老路径的中间卫星,使得切换很快得以实现。LQSR 算法的切换呼叫阻塞率低于 HPSR。分析表明,星际链路的拥塞情况、路径计算时长是影响切换呼叫能否成功完成的主要因素,LQSR 算法的负载均衡能力、计算复杂度均优于 HPSR。

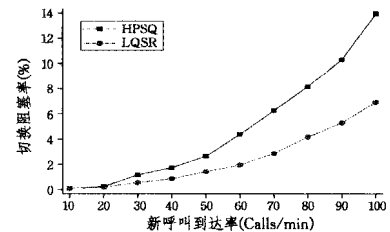


图 5 不同呼叫强度下切换呼叫阻塞率

### (4) 时延抖动

图 6 比较了呼叫强度对于 2 个算法的时延抖动性能的影响。随着呼叫强度的增加,HPSR 算法的时延抖动性能下降最为明显,这是因为 HPSR 仅寻找最短路径,而缺乏流量负载均衡机制,流量过于集中造成路径更容易拥塞,使得分组排队时间较长,引起时延抖动性能下降。而 LQSR 算法具有良好的负载均衡性能,避免了流量过于集中于某一区域。

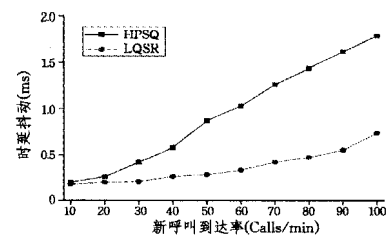


图 6 不同呼叫强度下时延抖动

**结束语** 本文引入移动代理技术,以 Walker 星座为研究对象,提出了一种轻量级按需 QoS 源路由算法 LQSR(Light-weight on-demand QoS Source Routing)。LQSR 算法选择具有最大覆盖时间的卫星作为接入卫星,当有呼叫到达时,源接入卫星派遣移动代理在网络中迁移,并建立数据传输路径;接入卫星发生切换时,启动路由重建算法更新数据传输路径。提出的移动代理迁移策略、星际 QoS 路由及其重建方法着眼于改善流量负载均衡、降低信令与计算开销。仿真结果表明,LQSR 的开销、呼叫阻塞率和时延抖动等 QoS 指标均优于传统路由算法。未来工作中将对拓扑变化更加复杂的极轨道星座 LEO 卫星网 QoS 路由由算法展开研究。

## 参考文献

- [1] Papapetrou E, Karapantazis S, Pavlidou F N. Distributed on-demand routing for LEO satellite systems[J]. Computer Networks, 2007, 51(15): 4356-4376
- [2] 唐剑, 余春东, 徐志明. LEO/MEO 卫星网络动态多径路由协议[J]. 计算机科学, 2009, 36(10): 42-45
- [3] 尹志忠, 张龙, 周贤伟, 等. 3 层体系结构卫星网络的性能分析[J]. 计算机工程, 2010, 36(12): 10-12
- [4] Zhou Y, Sun F, Zhang B. A novel QoS routing protocol for LEO and MEO satellite networks [J]. International Journal of Satellite Communications and Networking, 2007, 25(6): 603-617
- [5] 潘艳辉, 王韬, 李华. 非静止轨道卫星网络面向应用仿真分析[J]. 计算机仿真, 2011(8): 112-116
- [6] Nguyen H N, Jukan A. An Approach to QoS-Based Routing for LEO Satellite Networks[C]//Proceedings of IEEE Global Telecommunications Conference (GLOBECOM' 00). Vol. 2, San Francisco, CA, IEEE, 2000: 1114-1118
- [7] Xu Hui, Huang Fei, Wu Shi-qi. A distributed QoS routing based on ant algorithm for LEO satellite network [J]. Chinese Journal of Electronics, 2007, 24(6): 765-771
- [8] Karapantazis S, Papapetrou E, Pavlidou F N. Multiservice on-demand routing in LEO satellite networks[J]. IEEE Transactions on Wireless Communications, 2009, 8(1): 107-112
- [9] 肖甫, 孙力娟, 叶晓国, 等. 面向卫星网络的流量工程路由算法[J]. 通信学报, 2011, 32(5): 104-111
- [10] Huang Q, Yeo B S, Kong P Y. An enhanced QoS routing algorithm for provision of end-to-end delay guarantee in low earth orbit satellite networks[C]//Proceedings of 2005 IEEE Wireless Communications and Networking Conference (WCNC2005). Vol 3, New Orleans, USA: IEEE, March 2005: 1485-1490
- [11] Svigelj A, Mohoric M, Kandus G, et al. Routing in ISL networks considering empirical IP traffic[J]. IEEE Journal on Selected Areas in Communications, 2004, 22(2): 261-272
- [12] 朱军, 饶元, 傅雷扬, 等. 基于移动代理的卫星网路由性能研究[J]. 计算机工程与应用, 2012, 48(3): 69-72
- [13] Taleb T, Mashimo D, Jamalipour A. Explicit Load Balancing Technique for NGEOSatellite IP Networks with on-Board Processing Capabilities [J]. IEEE/ACM Transactions on Networking, 2009, 17(1): 281-293

(上接第 35 页)

b) 将一个 Simulink 模块添加到 UML 模型采用类似于“黑盒测试”的方法, 提取 Simulink 模块的输入、输出向量, 为 Simulink 模块增加数据流端口 (flowport)。数据流端口是 UML 中定义的模块交互接口。图 8 所示是封装输入输出接口的 Simulink 模块。

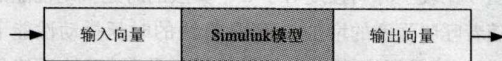


图 8 封装为 UML 模块的 Simulink 模型

c) 在 Rhapsody 中的 Simulink 模块之间的交互或是 Simulink 模块和其他 UML/Rhapsody 模块之间的交互与普通的 UML/Rhapsody 模块的流端口之间的交互方式一样, 都是采用链接器 (link connectors) 来实现, 链接器是 UML 定义的模块间交互方法。

d) 包含有 Simulink 模型的 Rhapsody 模块生成代码时, Simulink 模型通过调用 RTW 来实现代码生成, UML/Rhapsody 模块则通过 Rhapsody 自带的代码生成工具来实现, RTW 生成的代码则被包裹在 UML 生成的代码之中。

**结束语** 本文首先在分析 CPS 系统的特征和属性基础上将 CPS 系统划分为物理实体、计算实体和交互实体, 再通过精化、提取将 CPS 模型简化, 将系统划分为物理实体和计算实体。采用不同实体不同建模方式的思想, 分别提出了针对物理实体的基于连续时间的运动方程行为模型和面向计算实体的基于事件驱动的有限状态机行为模型。然后通过扩展物理实体行为模型和计算实体行为模型, 实现了物理模型信息化和计算模型物理化, 使两种模型融合在一起协同工作, 刻画 CPS 系统。最后对两种模型所能采用的建模工具进行了可行分析, 提出了物理行为模型采用 Simulink/RTW 来构建和生成代码, 计算实体模型采用统一建模语言 (UML) 进行建模, 并在此基础上提出了将 Simulink 模型转化为 UML 模型

方法, 完成了两种异质模型的协同建模仿真。

## 参考文献

- [1] 何积丰. Cyber-Physical-System [J]. 中国计算机学会通讯, 2010, 6(1): 25-29
- [2] Derler P, Lee E A, Sangiovanni-Vincentelli A L. Addressing modeling challenges in cyber-physical systems[R]. UCB/EECS-2011-17. EECS Department, Mar. 2011
- [3] Lin Jing, Sedigh S. A General Framework for Quantitative Modeling of Dependability in Cyber-Physical Systems; A Proposal for Doctoral Research[C]//33rd Annual IEEE International Computer Software and Applications Conference. 2009
- [4] Huang J, Bastani F. Extending Service Model to Build an Effective Service Composition Framework for Cyber-Physical Systems[J]. Service-Oriented Computing and Applications (SOCA), 2009, 1(1-8): 14-15
- [5] Lee E A. Cyber Physical Systems; Design Challenges[C/OL]. International Symposium on Object /Component /Service-Oriented Real-Time Distributed Computing (ISORC). 2010-03-22
- [6] Lee E A. Computing Foundations and Practice for Cyber-Physical Systems; A Preliminary Report[R]. No. UCB/EECS-2007-72. 2007
- [7] 杨新, 王小虎, 等. 飞机六自由度模型及仿真研究[J]. 系统仿真学报, 2000, 12(5): 210-213
- [8] 李文强, 彭学锋, 郑志强. 基于 Simulink 的无人机电六自由度仿真[J]. 系统仿真学报, 2007, 19(19): 4604-4606
- [9] Hooman J, Mulyar N, Posta L. Coupling Simulink and UML models[C]//Proc. Symposium (FORMSIFORMATS). 2004: 304-311
- [10] 沙静, 蒋昌俊. 基于 UML 实时状态图的定量分析[J]. 系统仿真学报, 2008, 20(8): 1957-1960
- [11] Farkas T, Meiseki E, Neumann C. Intergration of UML with Simulink into Embedded Software Engineering[C]//ICROS-SICE International Joint Conference 2009. 2009: 474-479