

# 具有服务质量保障的片上网络路由仲裁控制

官 铮<sup>1</sup> 钱文华<sup>1</sup> 杜长青<sup>2</sup>

(云南大学信息学院 昆明 650091)<sup>1</sup>

(云南曲靖师范学院计算机科学与工程学院 曲靖 655000)<sup>2</sup>

**摘要** 片上网络(Network on Chip, NoC)通过仲裁机制控制各个端口间的数据交换。经典的RR(Round-Robin)算法保障了各端口间的公平性,但却难以以为时延敏感业务提供具有低数据交换等待时延的QoS保障。为此,提出一种区分优先级并行调度RR(PP-RR)算法,其根据端口数据负载提供区分优先级服务,高优先级端口数据享有更多仲裁成功机会。采用并行调度两级轮控制模型实现了对PP-RR仲裁算法的数学建模,获得了时延性能的数学解析。

**关键词** 片上网络,仲裁,区分优先级,平均等待时延

中图分类号 TN929 文献标识码 A DOI 10.11896/j.issn.1002-137X.2015.2.012

## On-chip Network Router Arbitration Control with QoS Support

GUAN Zheng<sup>1</sup> QIAN Wen-hua<sup>1</sup> DU Chang-qing<sup>2</sup>

(School of Information Science and Technology, Yunnan University, Kunming 650091, China)<sup>1</sup>

(School of Computer Science and Engineering, Qujing Normal University, Qujing 655000, China)<sup>2</sup>

**Abstract** For the routers of network on chip (NoC), arbitrator plays an important role in the performance of packets switching between each port. The traditional round robin (RR) algorithm provides fairness among different ports. However, it seems difficult to guarantee QoS in delay characteristics. For this reason, a priority based parallel round robin (PP-RR) algorithm was proposed, which aims to provide differentiated service according to the communication traffic load. The high priority port will acquire more transmission opportunity in the arbitration. Furthermore, a mathematic analytic model based on the two-level polling systems was utilized for the performance evaluation of PP-RR algorithm. The closed form expression of mean waiting time is achieved.

**Keywords** NoC, Arbitrator, Priority differentiation, Mean waiting time

## 1 引言

片上系统(System on Chip, SoC)将计算资源即多个处理器和可重用的组件集成到一个芯片上,将计算资源之间的点到点的连接转向了总线的连接,使芯片的计算和处理能力显著提高。然而,基于总线的连接方式也使得SoC的设计面临功耗带宽限制、信号集成度、信号延迟以及全局同步等方面的问题。片上网络(Network on Chip, NoC)将计算机网络技术移植到片上系统设计中以解决片上组件的通信问题,实现了计算资源与通信结构(网络)的分离<sup>[1-3]</sup>。

2D Mesh是最常见的NoC拓扑结构,由资源节点、路由节点和通道构成。由IP内核或存储器系统构成的资源节点产生数据,经通道与路由节点连接。路由节点负责执行通信任务,通过设计仲裁机制对各个端口所发出的请求进行响应,是影响NoC网络延迟和吞吐量等性能的重要因素之一。大部分的NoC路由仲裁设计中都采用了RR算法来控制各个端口的请求响应,各端口的请求被平均轮询响应有效避免了

冲突;然而该算法无法有效区分业务优先级,当路由节点各端口存在流量差异,或流经某端口的业务需要更大带宽时则无法满足需求。对此,一些改进算法相继提出,但并未对关键性能参数进行数学解析。

本文采用嵌入式马尔可夫链和概率母函数的方法,针对基于RR的具有QoS保障的NoC路由器建立排队模型,完成对时延分析等关键性能参数的数学解析,通过仿真实验进行验证。

## 2 片上网络仲裁器

片上网络中,当来自不同方向多个数据包竞争同一个输出端口时,需要通过仲裁器选择一个数据成功通过输出端口,实现共享。典型的简单轮询(Round-Robin, RR)仲裁方式中,当前仲裁成功的请求端口在下一时钟周期中优先级最低,而来自其邻居端口的请求具有最高优先级,相当于输入端口被赋予一定的逻辑顺序,来自各个端口的数据按序依次占用输出端口。该仲裁方式最大的优点是保证了公平性且易于实

到稿日期:2014-03-04 返修日期:2014-06-23 本文受国家自然科学基金项目(61465051, 61463054),云南省应用基础研究计划青年项目(2012FD002),云南大学博士学位教师科研项目(XT412004)资助。

官 铮(1982-),女,博士,高级实验师,主要研究方向为无线传感器网络、网络接入技术、系统性能分析和优化等, E-mail: gz\_627@sina.com; 钱文华(1980-),男,博士,副教授,主要研究方向为计算机图形学、计算机网络;杜长青(1980-),男,硕士,讲师,主要研究方向为无线传感器网络、网络接入技术。

现<sup>[3]</sup>,但是,当片上网络中 IP 发送的信息不是同一种类型,以及各端口流量差异较大的情况下,绝对公平的仲裁方式则不适合,因此,需要一种支持区分服务质量的仲裁方式。轮询与优先级结合的仲裁方式兼顾了高效性和公平性,使那些流量较高(或实时业务)的端口相对于低流量(或普通业务)端口具有更多的响应机会,减少了数据堵塞,从而使整个网络的工作效率得到提高。

针对普通轮询仲裁器的缺陷,文献[4]提出了一种静态优先级仲裁器,它可以按固定的优先级对输入的请求进行仲裁。文献[5]提出了一种动态优先级仲裁器,它可以根据输入数据包的优先级安排输出授权。文献[6]在 NOC 路由节点中采用基于 Lottery 算法的仲裁机制,使流量大的端口具有更多的“中奖”机会,使较繁忙的信道获得更多的资源。文献[7]中通过建立队列模型分析了单汇聚点的 NoC 时延特性。朱樟明等设计了先锁存后仲裁方式的仲裁器,使一组请求中的高优先级请求率先得到授权,低优先级的请求后得到仲裁,相同优先级的请求顺序得到仲裁<sup>[8,9]</sup>。文献[10]中提出了一种基于循环的优先级调度算法,其以一轮循环为单位进行的,每轮循环路由器输入缓存的数据都是按照优先级顺序而不是 FCFS 通过交叉开关。大量文献针对传统仲裁方式进行了改进并通过仿真实验验证其效果,但鲜有文献对所提出的仲裁方式进行数学建模及性能评估。

Bao 建立的两级轮询模型用于 NoC 仲裁机制分析<sup>[11,12]</sup>,其将端口划分为两类优先级,通过区分服务策略以及查询顺序为高优先级端口提供更多的响应机会,文献[11]分析了普通端口采用多级门限服务时的平均排队队长特性,文献[12]分析了普通端口采用限定( $k=1$ )服务时的平均排队队长和平均等待时延特性,但上述基于两级轮询仲裁方式中由于切换至普通端口时要经过一个转换时间,在此期间无法进行数据交换,因此存在一定的时延浪费。

鉴于以上因素,为保障高优先级端口的传输速率,同时避免普通端口出现饥饿现象,本文在两级轮询策略的基础上增加并行调度机制,提出数据发送和端口查询的并行处理策略:当前仲裁成功的端口输出数据时,仲裁器查询轮询表确认下一发送端口,一旦当前端口完成数据输出,下一端口立即开始输出数据;当前端口无数据请求,经过一个查询时间后,来自轮询表中下一端口的数据被输出。仲裁器把对当前端口的请求响应与下一端口查询并行处理,以减小端口切换过程中查询转换时间造成的等待时间,在区分端口优先级的同时提供时延保证。

### 3 区分优先级并行调度 RR(PP-RR)

仲裁机制是影响 NoC 系统性能的重要因素之一。本文提出区分优先级并行调度 RR(PP-RR)仲裁机制。仲裁器将数据输入端口区分为一个高优先级端口,标记为  $Port_h$  和 4 个普通端口,以逻辑顺序标记为  $Port_i (i=1,2,3,4)$ 。在仲裁顺序上高优先级端口具有抢占性,每次普通端口仲裁成功完成输出后,若高优先级端口有数据等待便优先获得输出端口占用权。高优先级端口仲裁成功后输出所有排队数据直至缓存为空;普通端口依次仲裁成功,且每次仅允许输出一个排队数据。由此实现端口优先级的区分。另一方面,将端口切换判断与数据输出同时进行,按照轮询表,当高优先级端口完成数据输出后,下一普通端口立即开始输出数据包;若此端口无

数据请求,则经过一个查询时间后仲裁器按轮询表将数据交换权转至下一普通端口,从而减少查询时间的浪费,降低时延。

当然,按照上述控制机理,PP-RR 中要完成  $N$  个端口的一轮仲裁,最多情况下需进行  $2N-1$  次端口切换,相较于 RR 算法的  $N$  次切换有所增加,因此对硬件要求稍有提高。

图 1 所示为假设本地端口为高优先级端口时的一个实例,路由器上设计 5 个双向端口:北(North, N)、南(South, S)、东(East, E)、西(West, W)和本地(Local, L)。L 仲裁成功后按完全服务输出所有排队数据 L1 和 L2,同时仲裁器判断 N 为下一响应端口;普通端口 N 在 L 完成输出后按限定( $k=1$ )服务输出一个排队数据 N1;随后 L 再次获得输出端口占有权输出 L3,各端口按  $L \rightarrow N \rightarrow L \rightarrow S \rightarrow L \rightarrow E \rightarrow L \rightarrow W \rightarrow L$  依次仲裁成功。

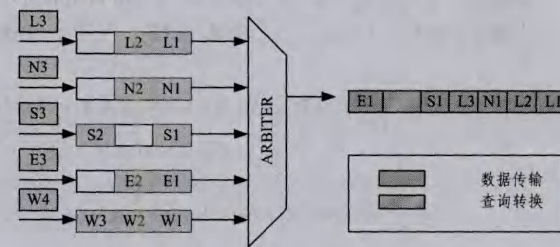


图 1 本地端口为高优先级端口的仲裁实例

### 4 PP-RR 建模和性能分析

文献[13]中我们讨论了一种离散时间并行调度两级轮询模型,在此利用该模型完成 PP-RR 仲裁的数学建模及性能分析。

#### 4.1 系统建模

定义随机变量  $\xi_i(n)$  为  $t_n$  时刻各方向普通端口  $Port_i (i=1,2,3,4)$  中等待传输的数据包数,发送请求获得响应后,普通端口输出 1 个数据包。高优先级端口  $Port_h$  的发送请求在  $t_n^*$  时刻得到响应,定义随机变量  $\xi_h(n^*)$  为  $t_n^*$  时刻  $Port_h$  中等待发送的数据包数量。高优先级端口按完全服务策略完成数据发送后,  $Port_{i+1}$  在  $t_{n+1}$  时刻获得响应,则  $t_n < t_n^* < t_{n+1}$ 。系统具有此时的状态变量  $\{\xi_1(n), \dots, \xi_4(n), \xi_h(n^*), \{\xi_1(n^*), \dots, \xi_4(n^*), \xi_h(n^*)\}$  和  $\{\xi_1(n+1), \dots, \xi_4(n+1), \xi_h(n+1)\}$ 。

假设数据包按独立同分布的泊松过程进入各端口缓存中排队等待响应,对于普通端口和高优先级端口其概率分布的母函数、均值和二阶特性分别为  $A_i(z_i), \lambda_i = A_i'(1), A_i''(1)$  和  $A_h(z_h), \lambda_h = A_h'(1), A_h''(1)$ 。端口发送一个数据包响应的时间服从于一个相互独立、同分布的概率分布,对于普通端口和高优先级端口其概率母函数、均值和二阶特性分别为  $B_i(z_i), \beta_i = B_i'(1), B_i''(1)$  和  $B_h(z_h), \beta_h = B_h'(1), B_h''(1)$ 。无数据响应时,由  $Port_i$  转向  $Port_h$  的查询转换时间服从于一个相互独立、且同分布的概率分布,其概率母函数、均值和二阶特性分别为  $R_i(z_i), \gamma_i = R_i'(1)$  和  $R_i''(1)$ 。

系统工作在稳定条件  $\sum_{i=1}^4 \rho_i + \rho_h < 1 (\rho_i = \lambda_i \beta_i, \rho_h = \lambda_h \beta_h)$  下,具有唯一的稳态分布。在  $t_n$  时刻,系统状态概率分布为:

$$\lim_{n \rightarrow \infty} P\{\xi_j(n) = x_j; j=1,2,3,4,h\} = \pi_i(x_1, \dots, x_i, \dots, x_4, x_h)$$

运用嵌入式马尔可夫链和概率母函数的分析方法,可建立系统状态概率母函数:

$$G_i(z_1, \dots, z_4, z_h) = \sum_{x_1=0}^{\infty} \dots \sum_{x_i=0}^{\infty} \dots \sum_{x_4=0}^{\infty} \sum_{x_h=0}^{\infty} z_1^{x_1} \dots z_i^{x_i} \dots z_4^{x_4} z_h^{x_h} \pi_i$$

$$(x_1, \dots, x_i, \dots, x_4, x_h), i=1, 2, \dots, N \quad (1)$$

根据模型服务规则,存在如下关系式:

$$\begin{cases} \xi_j(n+1) = \xi_j(n^*) + \eta_j(\nu_h), j \neq h \\ \xi_j(n+1) = 0, j = h \\ \xi_j(n^*) = \xi_j(n) + \eta_j(\nu_i), j \neq i, \xi_i(n) \neq 0 \\ \xi_j(n^*) = \xi_j(n) + \eta_j(\nu_i) - 1, j = i, \xi_i(n) \neq 0 \\ \xi_j(n^*) = \xi_j(n) + \mu_j(u_i), j \neq i, \xi_i(n) = 0 \\ \xi_j(n^*) = \mu_j(u_i), j = i, \xi_i(n) = 0 \end{cases}$$

利用上式计算得出  $t_{n+1}$  时刻系统状态变量的概率母函数:

$$G_{i+1}(z_1, \dots, z_4, z_h) = \lim_{n \rightarrow \infty} E \left[ \prod_{j=1}^4 z_j^{\xi_j(n+1)} z_h^{\xi_h(n+1)} \right]$$

$$= G_{ih}(z_1, \dots, z_4, B_h \left( \prod_{j=1}^4 A_j(z_j) F_h \left( \prod_{j=1}^4 A_j(z_j) \right) \right)) \quad (2)$$

$t_n^*$  时刻系统状态变量的概率母函数:

$$G_{ih}(z_1, \dots, z_4, z_h) = \lim_{n \rightarrow \infty} E \left[ \prod_{j=1}^4 z_j^{\xi_j(n^*)} z_h^{\xi_h(n^*)} \right]$$

$$= \frac{1}{z_i} B_i \left( A_h(z_h) \prod_{j=1}^4 A_j(z_j) \right) [G_i(z_1, \dots, z_i, \dots, z_4, z_h) - G_i(z_1, \dots, z_i, \dots, z_4, z_h) |_{z_i=0}] + R_i \left( A_h(z_h) \prod_{j=1}^4 A_j(z_j) \right) G_i(z_1, \dots, z_i, \dots, z_4, z_h) |_{z_i=0} \quad (3)$$

## 4.2 时延特性分析

定义  $g_i(j)$  表示服务器在  $t_n$  时刻轮询普通端口时,端口  $j$  缓冲区内平均等待的顾客数(平均排队队长),服务器在  $t_n^*$  时刻轮询高优先级端口时,端口  $j$  缓冲区内平均等待的顾客数为  $g_{ih}(j)$ 。

定义  $g_i(j)$  和  $g_{ih}(j)$  如下:

$$g_i(j) = \lim_{z_1, \dots, z_i, \dots, z_4, z_h \rightarrow 1} \frac{\partial G_i(z_1, \dots, z_j, \dots, z_4, z_h)}{\partial z_j},$$

$$i=1, \dots, 4; j=1, \dots, 4, h \quad (4)$$

$$g_{ih}(j) = \lim_{z_1, \dots, z_i, \dots, z_4, z_h \rightarrow 1} \frac{\partial G_{ih}(z_1, \dots, z_j, \dots, z_4, z_h)}{\partial z_j},$$

$$i=1, \dots, 4; j=1, \dots, 4, h \quad (5)$$

以及:

$$g_i(j, k) = \lim_{z_1, \dots, z_i, \dots, z_4, z_h \rightarrow 1} \frac{\partial^2 G_i(z_1, \dots, z_j, \dots, z_k, \dots, z_4, z_h)}{\partial z_j \partial z_k},$$

$$i=1, 2, \dots, N; j=1, \dots, 4, h; k=1, \dots, 4, h \quad (6)$$

$$g_{ih}(j, k) = \lim_{z_1, \dots, z_i, \dots, z_4, z_h \rightarrow 1} \frac{\partial^2 G_{ih}(z_1, \dots, z_j, \dots, z_k, \dots, z_4, z_h)}{\partial z_j \partial z_k},$$

$$i=1, \dots, 4; j=1, \dots, 4, h; k=1, \dots, 4, h \quad (7)$$

$$1 - G_i(z_1, \dots, z_i, \dots, z_4, z_h) |_{z_i=0} = \lambda_i \bar{\theta}, i=1, \dots, 4 \quad (8)$$

将式(2)、式(3)代入式(4)、式(5)计算,化简后结合式(8)

得出:

$$\bar{\theta} = \frac{1 - G_i(z_1, \dots, z_i, \dots, z_4, z_h) |_{z_i=0}}{\lambda_i} = \frac{4\gamma}{1 - \rho_h - 4\rho + 4\lambda\gamma} \quad (9)$$

基于我们在文献[13]中的分析结果可知,高优先级端口的平均排队队长  $g_{ih}(h)$  可由式(2)、式(3)代入式(4)、式(5)计

算得到,如式(10)所示:

$$g_{ih}(h) = \frac{\lambda_h \gamma (1 - \rho_h)}{1 - \rho_h - 4\rho + 4\lambda\gamma} \quad (10)$$

普通端口平均排队分组数由式(2)、式(3)代入式(6)、式(7)计算得到,结果经化简后得到  $g_i(i)$  如下表达式:

$$g_i(i) = \frac{1 - \rho_h}{2(1 - N\rho - \rho_h)} \left\{ N(\gamma A''(1) + \lambda^2 R''(1)) + \frac{2N\rho_h}{1 - \rho_h} (\lambda^2 R''(1) + \lambda^2 r) + \frac{N\rho_h^2}{(1 - \rho_h)^2} (\gamma A''(1) + \lambda^2 R''(1)) + \frac{N\lambda}{1 - N\rho - \rho_h + N\lambda\gamma} [N\gamma(\beta - \gamma)A''(1) - N\lambda^2 \gamma R''(1) + \frac{1}{1 - \rho_h} (2N\lambda^2 \gamma \rho_h (\beta - \gamma) - 2N\lambda^2 \gamma (1 + \rho_h) R''(1) + 2\gamma(1 - \rho - \rho_h) + \lambda\gamma \rho_h (1 + \rho_h) - (N-1)\lambda\gamma^2 + 2(N-1)\lambda\gamma(\gamma - \beta)) + \frac{\lambda}{(1 - \rho_h)^2} (N\lambda\gamma B''(1) - N\lambda\gamma \rho_h^2 R''(1) + \gamma \rho_h^2 (\rho_h + N\lambda(\beta - \gamma))A''(1) + \lambda_h \gamma B_h''(1))] \right\}, N=4 \quad (11)$$

由式(3)代入式(7)计算,  $g_{ih}(h, h)$  表达式如下:

$$g_{ih}(h, h) = [A''_h(1)\beta + \lambda_h^2 B''(1) - \lambda_h^2 R''(1) - A''_h(1)\gamma] \lambda \bar{\theta} + \lambda_h^2 R''(1) + A_h''(1)\gamma \quad (12)$$

数据包等待时延  $w_j$  定义为该数据包进入  $port_j$  到其发送请求得到响应的间隔时间,  $E[w_i]$  和  $E[w_h]$  分别定义为普通端口和高优先级端口中数据包的平均等待时延。

高优先级端口按完全服务策略完成请求响应,分析计算得到高优先级端口数据包平均等待时间:

$$E[w_h] = \frac{g_{ih}(h, h)}{2\lambda_h g_{ih}(h)} - \frac{A''_h(1)}{2\lambda_h^2 (1 + \rho_h)} + \frac{\lambda_h B_h''(1)}{2(1 - \rho_h)} \quad (13)$$

将式(10)和式(12)代入式(13)可得到高优先级端口平均等待时间理论计算值。

普通端口按并行限定( $k=1$ )服务方式,每次获得响应后仅能发送一个数据包。分析计算得到普通端口数据包平均等待时间:

$$E[w_i] = \frac{1}{\lambda C} g_i(i) - \frac{1}{\lambda} - \frac{A''(1)}{2\lambda^2} \quad (14)$$

将式(11)代入上式可计算得到普通端口平均等待时理论计算值。

## 4.3 稳定工作条件

各端口缓冲中数据包的到达及输出相当于一个生灭过程,在轮询控制模式下,要求所有端口的总负载  $G < 1$ , 否则排队队长将逐渐趋于无穷大而无法收敛。实际应用中随着  $G$  增加到一定程度,系统平均排队队长、平均等待时延等会急剧增加,为保证服务性能需将负载控制在一定范围内。

在 RR 算法和文献[11]中,由于端口服务(文献[11]中的普通端口)采用了限定( $k=1$ )服务方式,且端口切换时需耗费查询转换时间  $r$ , 因此 PP-RR 控制算法中在请求端口有数据等待的情况下无需查询转换时间。端口总负载需满足如下关系:

$$\text{RR 算法: } G|_{RR} = \sum_{i=1}^5 \lambda_i (\beta_i + \gamma_i)$$

$$\text{文献[11]: } G|_{Ref[11]} = \sum_{i=1}^4 \rho_i + \sum_{i=1}^4 \lambda_i \gamma_i + \rho_h$$

$$\text{PP-RR 算法: } G|_{PP-RR} = \sum_{i=1}^4 \rho_i + \rho_h$$

由上式可知在相同端口数据参数下,  $G|_{PP-RR} \leq G|_{Ref[11]} \leq G|_{RR}$ , 因此 PP-RR 算法具有更大的稳定工作范围。

## 5 数值分析

为了进一步验证理论分析的正确性和本文所述模型的有效性,本节基于 NIRGAM 对所设计的路由节点仲裁算法分析模型进行了验证和评估,并将其与 RR 仲裁算法和文献[11]两级轮询模型进行了比较。仿真频率为 1GHz,节点工作在一般模式,即每个源节点以相同的概率随机地向其他节点发送数据,数据包之间的时间间隔由数据到达率  $\lambda$  确定。理论计算和模拟实验采用相同的参数值,其中 Theory 为采用本文所建模型对 PP-RR 算法平均等待时延的理论计算值,Simulation 为 PP-RR 仲裁策略计算机仿真结果。仿真过程中通过 MATLAB 产生 Poisson 分布随机序列模拟数据到达过程。

表 1 为 PP-RR 仲裁算法下平均响应周期随负载变化的比较, $\Sigma\rho$  和  $\rho_h$  ( $\rho = \lambda\beta, \rho_h = \lambda_h\beta_h$ ) 分别表示归一化后的系统总负载和高优先级端口负载。如表 1 所列,本文建立数学模型的理论计算与仿真具有一致性。

表 1 平均循环周期

|            | $\rho_h=0.1$                         | $\rho_h=0.2$ | $\rho_h=0.3$ | $\rho_h=0.4$ | $\rho_h=0.5$ |
|------------|--------------------------------------|--------------|--------------|--------------|--------------|
|            | $(\rho_i=0.1, \beta_h=\beta=2, r=1)$ |              |              |              |              |
| Theory     | 5.7143                               | 6.6667       | 8            | 10           | 13.3333      |
| Simulation | 5.6232                               | 6.7501       | 7.5896       | 10.4026      | 13.0528      |

表 2 为 PP-RR 仲裁算法下各端口数据包平均等待时延理论计算值与仿真结果的比较,如表 2 所列,本文建立数学模型的理论计算与仿真具有一致性。证明通过理论分析解析式对协议性能评估切实可行。

表 2 平均等待时延

|            | Queue h                                | Queue 1 | Queue 2 | Queue 3 | Queue 4 |
|------------|--|---------|---------|---------|---------|
|            | $(\rho=0.1, \rho_h=0.2, \beta=2, r=1)$ |         |         |         |         |
| Theory     | 0.5000                                 | 5.7500  | 5.7500  | 5.7500  | 5.7500  |
| Simulation | 0.3977                                 | 5.8693  | 6.3254  | 5.4668  | 5.2146  |

图 2 和图 3 中,我们将本文所述 PP-RR 仲裁策略与 RR 和文献[11]进行了比较。通过表 1 和表 2 我们已经验证了理论分析的正确性,因此图 2、图 3 中的数据直接通过理论计算获得。

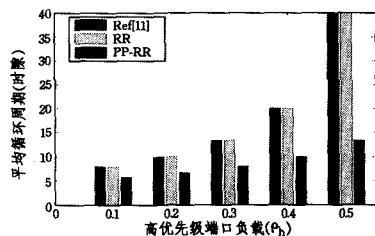


图 2 平均循环周期 ( $\rho_i=0.1, \beta_h=\beta=2, r=1$ )

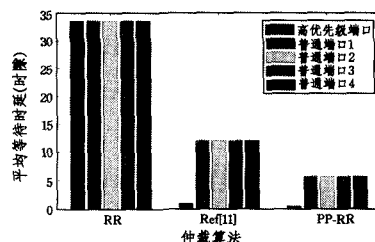


图 3 平均等待时延 ( $\rho_h=0.2, \rho_i=0.1, \beta_h=\beta=2, r=1$ )

图 2 为不同仲裁算法下平均响应周期随负载变化的比

较, $\Sigma\rho$  和  $\rho_h$  ( $\rho_h = \lambda_h\beta_h$ ) 分别表示归一化后的系统总负载和高优先级端口负载。PP-RR 算法对各端口响应一轮的平均循环周期较小。该趋势在高负载时更加明显: $\rho_h=0.1$  ( $\Sigma\rho=0.5$ ) 时,PP-RR 平均循环周期为 RR 算法的 71%, $\rho_h=0.5$  ( $\Sigma\rho=0.9$ ) 时,PP-RR 平均循环周期仅为 RR 算法的 33.2%,原因是 RR 仲裁方式以及文献[11]所述控制策略中,无论端口是否输出数据都要经过一个转换时间后下一端口才能开始发送,而本文所述方法中,端口切换和数据输出并行处理,低负载时,由于端口空闲概率较高,此时端口的切换仍然需要消耗转换时间;随着负载增加,端口空闲切换的概率降低,当前端口输出数据后可立即切换至下一端口,有效减小了平均循环周期。

图 3 为不同仲裁算法下各端口数据包平均等待时延的比较。仿真过程中假设高优先级端口流量较高,参数设置归一化后高优先级端口负载分别为 0.2,其余 4 个普通端口流量均衡,负载为 0.1。观察图 3 数据表明时延特性方面: $\Sigma\rho=0.6$  时,PP-RR 高优先级端口数据平均等待时延为 RR 算法的 1.6%,文献[11]的 52.1%;普通端口数据平均等待时延仅为 RR 算法的 17.2%,文献[11]的 47.9%;另外,在系统稳定性方面:当  $\rho_h=0.2, \rho_i=0.1, \Sigma\rho=0.6$  时,RR 和文献[11]算法  $G>1$ ,均已超出稳定范围,而 PP-RR 仍能保障数据稳定传输。

上述比较结果表明,PP-RR 为基于轮询的仲裁方式保证了公平性,此外,通过为高优先级端口分配更多的资源实现了优先级区分,并且数据发送和端口查询的并行处理,有效提高了网络的利用率,减小端口排队数据包数量,增加了稳定工作区间,降低了等待时延,有助于系统提高时延性能。

**结束语** 本文在基本 RR 算法的基础上,建立区分优先级的并行调度 RR(PP-RR)路由节点仲裁算法,通过调度顺序和服务方式等方面的改进以提高网络效率、减小数据发送等待时延。此外,采用嵌入式马尔可夫链和概率母函数的方法,对 PP-RR 建立排队模型,完成对各端口数据包输出平均等待时延等关键性能参数的精确数学解析。数学分析过程中假设普通端口数据包到达率服务、数据包服务时间相同,对具有不同到达率和服务时间的端口进行分析将具有更广泛的应用范围,这将是下一步的研究重点。

## 参考文献

- [1] Henkel J, Wolf W, Chakradhar S. On-chip Networks a Scalable Communication-centric Embedded System Design Paradigm [C]// Proceedings of the 17th International Conference on VLSI design. Piscataway, NJ, USA: IEEE Press, 2004: 845-851
- [2] Liu J, Zheng L R, Tenhunen H. Interconnect Intellectual Property for Network-on-Chip [J]. Journal of Systems Architecture, 2004, 50(2/3): 65-79
- [3] Shin E S, Mooney V J, Riley G. Round-robin Arbiter Design and Generation [R]. Technical Report GIT-CC-02-38. College of Computing, Georgia Institute of Technology, July 2002
- [4] Felicima F, Furber S B. An Asynchronous On-chip Network Router with Quality-of-Service (QoS) Support [C]// Proceedings of IEEE International SOC Conference. Santa Clara: IEEE, 2004: 274-277
- [5] Zid M, Zitouni A, Bagmane A, et al. New Genetic GALS NoC Architectures with Multiple QoS [C]// International Conference on Design and Test of Integrated Systems in Nanoscale Tech-

- [6] 武畅,李玉柏,彭启琮,等.可设置仲裁优先程度的 NOC 路由节点设计[J].电子科技大学学报,2008,37(5):645-648
- [7] Benny V H. Numerical solution of polling systems for analyzing networks on chips[C]//Proceedings of NSMC 2010. Williamsburg, USA, 2010; 90-93
- [8] 管旭光,杨银堂,朱樟明.用于片上网络的准时延不敏感全异步仲裁器[J].西安电子科技大学学报:自然科学版,2011,38(3):83-89
- [9] 杨延飞,朱樟明,周端,等.用于片上网络的延时无关异步动态优先级仲裁器[J].西安电子科技大学学报:自然科学版,2012,39(1):42-48
- [10] Lu C H, Chiang K C, Hsiung P A. Round-based priority arbitra-

tion for predictable and reconfigurable Network-on-Chip[C]//International Conference on Field-Programmable Technology, 2009(FPT 2009). 2009

- [11] Liu Qian-lin, Zhao Dong-feng, Zhao Yi-fan. An efficient priority service model with two-level-polling scheme [J]. High Technology Letters, 2011, 17(3): 245-251
- [12] Bao Li-yong, Zhao Dong-feng, Zhao Yi-fan. A Priority-based Polling Scheduling Algorithm for Arbitration Policy in Network on Chip [J]. Journal of electronics (China), 2012, 29(1/2): 120-127
- [13] Guan Zheng, Zhao Dong-feng, Zhao Yi-fan. A Discrete Time Two-level Mixed Service Parallel Polling Model [J]. Journal of electronics (China), 2012, 29(1/2): 103-110

(上接第 42 页)

分布趋于随机,则 Internet 网络稳定性不再受地理因素影响;(2)提高链路铺设和维护技术水平,使得路由器间建立距离增大,铺设维护代价减小,不仅能够提高效率,而且还能够增强整个 Internet 网络的稳定性。

**结束语** AS 网络是一个处在开放环境中,并和环境中的其他多种网络系统相互作用的网络。各种客观因素对 AS 网络拓扑特性的影响是网络系统间的相互作用的重要体现。因此,本文提出地理超图结构来刻画 AS 网络的地理环境因素,进而提出地理超图和 AS 网络相互作用的机制,建立了 AS 的双层网络模型。该模型弥补了以往模型未能正确刻画 AS 地理分布的缺陷,使探索地理因素和 AS 网络性能的联系成为可能。另外,与仅关注 AS 网络现状和发展趋势的传统研究角度不同,本文侧重研究受到经济宏观调控或者技术突破等因素影响后 AS 网络的性能变化,为制定宏观调控措施和把握网络技术导向提供了理论依据。

## 参 考 文 献

- [1] Faloutsos M, Faloutsos P, Faloutsos C. On power-law relationships of the Internet topology[J]. ACM SIGCOMM Computer Communication Review, 1999, 29(4): 251-262
- [2] Siganos G, Faloutsos P, Faloutsos C. Power laws and the AS-level Internet topology[J]. IEEE/ACM Transactions on Networking, 2003, 11(4): 514-524
- [3] Bu T, Towsley D. On Distinguishing between Internet Power Law Topology Generators[J]. IEEE INFOCOM, 2002, 2: 638-647
- [4] Klemm K, Eguiluz V M. Highly Clustered Scale-Free Networks [J]. Physical Review E, 2002, 65(3)
- [5] Zoltán Toroczkai, Bassler K E. Jamming is limited in scale-free systems[J]. Nature, 2004, 428(15)
- [6] Albert R, Jeong H, Barabasi A L. Attack an error tolerance in complex networks[J]. APRIL, 2000, 406: 387-482
- [7] Winick J, Jamin S. Inet-3.0: Internet topology generator [R]. Technical report CSE-TR-456-02, Department of EECS, University of Michigan, 2002
- [8] Albert R, Barabasi A L. Topology of evolving networks: Local events and university[J]. Phys. Rev. E, 2004, 70: 066108
- [9] Medina A, Lakhina A, Matta I, et al. BRITE: An approach to universal topology generation [C]// Proceedings of MASCOTS, Washington, 2001: 346-353
- [10] Bu T, Towsley D. On distinguishing between Internet power law topology generators [C]// Proceeding of INFCOCOM, New-

York, 2002, 2: 638-647

- [11] Zhou S, Mondragon R J. Accurately modeling the Internet topology[J]. Phys. Rev. E, 2004, 70: 066108
- [12] Zhou S, Mondragon R J. Towards modeling the Internet topology-the interactive growth model[J]. Teletraffic science and engineering, 2003, 5: 121-130
- [13] Park S T, Pennock D M, Giles C L. Comparing static and dynamic measurements and models of the Internet's topology[C]// Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies, 2004, 3: 1616-1627
- [14] Sagy B, Mira G, Avishai W. An incremental super-linear preferential Internet topology model [C]// Proc 5th Annual Passive and Active Measurement Workshop. LNCS 3015, 2004: 53-62
- [15] Wang Xiao-ming, Loguinov D. Wealth-Based Evolution Model for the Internet AS-Level Topology [C]// Proceeding of INFOCOM, New York, 2006
- [16] Sagy B, Mira G, Avishai W. A geographic directed preferential Internet topology model[J]. Arxiv: CS, 2005, NI/0502061
- [17] Lakhina A, Byers J W, Crovella M, et al. On the geographic location of Internet resources (short abstract) [C]// Proc. SIGCOMM Internet Measurement Workshop (IMW'02). 2002: 249-250
- [18] Lakhina A, Byers J W, Crovella M, et al. On the geographic location of Internet resources[J]. IEEE Journal on Selected Areas in Communications, 2003, 21: 934-948
- [19] Yook S H, Jeong H, Barabasi A L. Modeling the Internet's large-scale topology [J]. Proc. Nat. Acad. Sci., 2002, 9(21): 13382-13386
- [20] Sterbenz J P G, Çetinkaya E K, Hameed M A, et al. Evaluation of network resilience, survivability, and disruption tolerance: analysis, topology generation, simulation, and experimentation [J]. Telecommunication systems, 2013, 52(2): 705-736
- [21] Willinger W, Roughan M. Internet Topology Research Redux [OL]. [http://sigcomm.org/education/ebook/SIGCOMMBook201301\\_chapter1.pdf](http://sigcomm.org/education/ebook/SIGCOMMBook201301_chapter1.pdf)
- [22] Aditya A, Shuchi C, Arvind K, et al. On the Scaling of Congestion in the Internet Graph [J]. ACM SIGCOMM Computer Communication Review, 2004, 34(3): 43-56
- [23] Tangmunarunkit H, Doyle J, Govindan R, et al. Does AS size determine degree in AS topology [J]. ACM SIGCOMM Computer Communication Review, 2001, 31(5): 7-10
- [24] Chvatal V. Tough graphs and hamiltonian circuits [J]. Discrete Math, 1973, 5: 215-228