

无数据缓存的容错环形 NoC

张丽果^{1,2} 杜慧敏² 韩俊刚^{1,2}

(西安电子科技大学微电子学院 西安 710071)¹ (西安邮电学院计算机系 西安 710121)²

摘 要 提出一种分层双组双环 NoC 拓扑结构,该结构中链路分为两组环网,其中有一组环网为主环,另一组为备用环网,用于 NoC 网络的容错。每组环网中包含一个控制环和一个数据环,控制环采用包的形式交换结点之间路由、链路错误和差错控制信息,数据环用电路交换方式进行数据通信。针对以上 NoC 拓扑结构,提出交换结点无需缓冲区的三级流水线结构,使得各个 IP 之间的数据通信延时最小。环网中采用时分复用和优先级相结合的机制,实现了公平路由和带宽的空分复用。仿真结果表明,该结构可以有效避免拥塞、死锁和饥饿,保证带宽充分利用,与理论分析一致。

关键词 片上网络,时分复用,流水线,空分复用,环网,容错

Fault-tolerance Ring Network on Chip without Buffer

ZHANG Li-guo^{1,2} DU Hui-min² HAN Jun-gang^{1,2}

(School of Microelectronics, Xidian University, Xi'an 710071 China)¹

(Computer Science Department, Xi'an Institute of Posts and Telecommunications, Xi'an 710061, China)²

Abstract The paper presented a hierarchical fault-tolerance ring NoC topology in which links are divided into two groups of rings. One group is used for fault-tolerance in case of the other group fails. Each group contains a control-ring and a data-ring. Control-ring propagates information of routing, link error and error control of all nodes by using packet and data-ring exchanges data by circuit switching. The paper proposed three-stage pipeline architecture without buffer which minimizes the delay of data communications among nodes. Ring network adopts Time-Division Multiplexing and priority mechanism to implement space-division-multiplexing of bandwidth and fair routing. Simulation results show that the architecture can avoid traffic congestion, deadlock and hungry and ensure use of bandwidth effectively. Simulation results coincident with the theory analysis.

Keywords Network on chip, Time-division multiplexing, Pipeline, Space division multiplexing, Ring, Fault-tolerance

1 引言

随着半导体工艺的不断发展和 SoC(System-on-Chip)技术的不断完善,SoC 中所包含的 IP 核数目不断增加。现有的以总线结构为通信基础的 SoC 技术面临着在性能、功耗、延时和可靠性等方面的巨大挑战^[1-3]。在 2001 年左右,一些研究机构借鉴和吸收了通信网络和并行计算的思想,提出了以通信为核心的复杂 SoC 的 IP 核集成方法,即片上网络(Network-on-Chip, NoC),采用包交换和路由的方法替代传统总线,实现处理单元(IP 核)与通信结构(网络)的分离^[4,5],以解决复杂 SoC 面临的各种问题。

由于 Mesh 拓扑结构的规则性和易扩展性,有利于布局布线,因此有关 NoC 的很多路由算法^[6]、容错^[7]、服务质量^[8]、功耗等问题^[9]的研究都是基于 Mesh 结构及其变型结构(如 Torus 结构)。其中,比较有影响的 Mesh NoC 系统是瑞典皇家工学院的 Nostrum^[10] 系统。

环网由于其拓扑结构简单,结点之间具有固定的跳数,路由简单,易于实现时分复用、空分复用和容错等诸多优点,在宏观网络中得到广泛应用。例如,世界范围内广泛使用同步数字体系 SDH^[8]以及目前正在研究的弹性分组环 RPR^[9]是典型的环网结构。

在 NoC 研究领域,环形拓扑结构受到了诸多的关注。芬兰 Tempere 大学开发的 Proteo^[10],采用分层双环结构,不同的业务具有不同结构,不同的业务有不同的服务质量。瑞典 Jönköping University 大学^[11]受城市交通线路启发,提出了环路的 R²NoC 结构,以解决路由由延时问题。这两种结构均支持包交换。

本文的研究背景是针对我国提出的 TD-SCDMA 无线通信标准,用 NoC 实现其基站子系统内部的通信。借鉴 SDH(Synchronous Digital Hierarchy)和 RPR(Resilient Packet Ring)中环网的思想,提出一种分层同步容错环形 NoC 拓扑结构。该结构中链路分为两组环网,其中有一组环网为主环,

到稿日期:2008-04-08 本文受 863 国家计划基金项目(2007AA01Z111),国家自然科学基金(90607008),西安应用材料创新基金(XA-AM-200615),陕西省教育厅科研资助(08JK429)资助。

张丽果 博士研究生,主要研究方向为电路与系统、NoC 设计与验证, E-mail:zhanglg@xyyou.edu.cn;杜慧敏 教授,主要从事 SoC 和 NoC 的形式化设计与验证;韩俊刚 教授,博士生导师,主要从事 SoC 的形式化设计与验证、计算机应用技术等研究。

另一组为备用环网,用于 NoC 网络的容错。每组环网中包含一个控制环和一个数据环,控制环采用包的形式交换结点之间路由信息、链路错误信息和差错控制信息,数据环用电路交换方式进行数据通信。为叙述方便,后面将主环称为正环,备用环称为反环。针对以上 NoC 拓扑结构,设计了交换结点无需缓冲区的三级流水线结构,使得各个 IP 之间的数据通信延时最小。环网中采用时分复用和优先级相结合的机制,实现了公平路由和带宽的空分复用。仿真结果表明,该结构可以有效避免拥塞、死锁和饥饿,保证带宽充分利用,与理论分析一致。

与文献[13]相比,有以下几点区别:1)本文设计的拓扑结构包括两组环,具有了容错功能。2)采用电路交换和包交换结合,通过流水线技术,无需数据交换节点中的缓冲区,降低了数据通信的延时。3)优先级和时分复用技术相结合,有效避免拥塞、死锁和饥饿。

2 空分和时分复用的分层环网 NoC 结构

借鉴 SDH 和 RPR 中环网的思想,我们提出一种分层的同步双组双环的 NoC 结构。在此 NoC 中环采用两层结构:第一层为主干环网结构,第二层称为子网络结构。子网络可以根据不同的应用,采用不同的结构形式。结点的编码也采用分层形式。图 1 是一种环网结构的典型例子。子网络可以包含线性结构、总线结构和星形结构。当然,子网中还可以包含其它结构,如 Mesh 或者环结构。一般而言,子网中的结点信息交换量比较大,而子网通过主干网进行数据交换的信息量相对较少。每组环包含了两个环,虚线标识的环为控制环,实线标识的环称为数据环。设置两组环的目的是提高 NoC 网络中链路的容错能力,容错的机制将在后面章节给出。该 NoC 采用包交换和电路交换相结合的方式,控制环中采用包交换,而数据环采用电路交换方式。

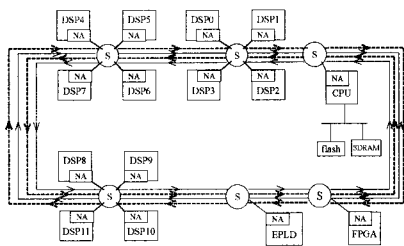


图 1 带有多个结点的环网结构

NoC 中每个交换结点分成 3 部分:第 1 部分用于处理控制环的信息,第 2 部分处理数据环的信息,第 3 部分用于处理与子网的接口。为叙述方便,后面我们将交换结点的 3 个部分分别称为控制结点、数据结点和子网接口结点。控制结点与其相连的环路构成控制环,数据结点与其相连环路构成数据环,子网通过子网接口结点与主干网相连接,因此可以把子网接口结点看成是子网的一部分。

每个控制结点的作用是:

- 1)根据子网接口结点需要发送的数据信息,向环网上所有控制结点广播需要建立的数据链路的信息。
- 2)根据收到广播信息,按照空分复用算法判断相应的数据结点是发送数据、转发数据、接收数据还是空闲(无数据)。
- 3)维护一张全局路由表,同步更新整个环路的链路信息。

3 环网 NoC 体系结构

3.1 流水线设计

为了解决拥塞、饥饿、死锁等问题,减少 IP 之间的数据通信延时,我们对图 1 的 NoC 结构设计了以下的控制结构:

1)数据环和控制环均采用时分复用技术,环中的每个结点都有两个循环计数器,分别对数据时隙和控制时隙进行计数,计数器的最大值就是环网中的结点数目 n 。整个环使用统一的复位和时钟信号,以保证环网的同步。把每个计数值当成一个时隙,分别将数据环和控制环上的时隙称为数据时隙和控制时隙。

2)若环网中结点数为 n ,则一个数据时隙包括 n 个控制时隙,可以用第 i 个数据时隙对应的 n 个控制时隙确定第 $i+1$ 个数据时隙数据环上的通信。

3)环中的每个控制结点都有一个优先级,占用下个数据时隙的结点具有最高优先级,其它结点的优先级依次降 1。优先级最高的结点可以优先占用环上的带宽,剩余带宽根据结点优先级不同进行动态分配,算法在 3.2 节中讨论。

采用流水线方式,在数据环进行数据通信之前的一个数据时隙,控制结点按照一定的算法计算出其对应数据结点是转发、接收、发送还是空闲。如果数据结点是空闲的,那么说明该结点在下一个数据时隙没有任何操作。如果是接收结点,那么下个数据时隙通过子网接口结点通知子网建立相应的接收路径。如果是转发结点,那么只完成数据的转发工作。

3.2 环网 NoC 中的相关算法

正常情况下备用数据环也用于传递数据,备用控制环不用。如果有一条数据环或控制环发生故障,那么可以用一条数据环或另外一条控制环进行通信,实现系统容错。为了充分利用链路的带宽,本系统采用带宽空分复用技术,多个源结点可以同时发送数据。在链路带宽有剩余的情况下,保证优先级高的结点可以获得更多的带宽。下面是对优先级机制及空分复用算法的描述。

1)结点编码与优先级:为链路上的每个结点分配一个二进制数,该二进制数代表了结点的编码,也代表了结点的优先级。这样,优先级与结点编码一致,优先级越高,结点编码越小。在比较结点优先级时,占用下个数据时隙结点的优先级变成最高优先级,其它结点的优先级自动降 1 后排列。例如,假设一个环路一共有 16 个结点,每个结点的编码依次为 0, 1, 2, ..., 15;如果占用下个数据时隙的结点为 3,则优先级排列顺序为 3, 0, 1, 2, 4, ..., 15。

2)路径:路径定义为从源结点到目标结点之间结点的有序集合。环网上第 p 个结点发送数据的目标地址为 q ,那么根据编码原则,第 p 个结点正向发送数据的结点集合为

$$set^p = \{addr | (p < addr < q) \wedge (p < q) \vee (addr > p \vee addr < q) \wedge (p > q)\}$$

反向发送数据的结点集合为

$$set^p = \{addr | (q < addr < p) \wedge (p > q) \vee (addr < p \vee addr > q) \wedge (p < q)\}$$

3)正环和反环的目标结点集合:假设在结点 p 之前的控制时隙中,在正向环上已经建立 i 条路径,在反向环上已经建立 j 条路径。定义 $D_i = \{e_i\}$ 为第 i 条路径结束元素的集合。在此定义正环目标结点集合:

$reward_den_set = \{node | node \in set^i \wedge i \leq p-1 \wedge node \in D_i\}$

反环目标结点集合:

$forward_den_set = \{node | node \in set^i \wedge j \leq p-1 \wedge node \in D_j\}$

且有 $reward_den_set \cap forward_den_set = \emptyset$ 。

4) 目标结点: 如果 $p \in reward_den_set$ 或者 $p \in forward_den_set$, 那么 p 结点是正环或者是反环目标结点。

5) 结点 p 在正环上转发的条件是: 假设集合 $A_i = \{s_i, e_i\}$ 为第 i 路径开始与结束元素的集合, 如果 $\exists m (m = p \wedge m \in U_i \{set^i - A_i\})$, p 是正环方向的转发结点; $\exists m (m = p \wedge m \in U_j \{set^j - A_j\})$, 那么 p 是反环方向上的转发结点。

6) 结点 p 在正环上可以发送数据的条件是: 如果 set^p 集合中所有的结点之间的正环方向的链路正常且 $\forall_j < P-1, set^p \cap set^j = \varphi$; 结点 p 在反环上可以发送数据的条件是: 如果 set^p 集合中所有的结点之间的反环方向的链路正常且 $\forall_j < p-1, set^p \cap set^j = \varphi$ 。

7) 带宽以正环优先分配: 也就是说, 在正数据环无法建立某个结点的通信路径, 然后在反环上建立通信路径。

这种动态分配带宽的算法有以下优点:

1) 采用时分复用的方法, 保证每个结点都有机会发送数据, 因此保证了环路每个结点不会产生饥饿的现象。

2) 采用带宽空分复用, 多个源结点可以同时发送数据。在链路有空闲的情况下, 保证优先级高的结点有更多的带宽。图 2 表示出了环上的结点之间的通信链路。如图 2 所示, 结点 3 占用发送时隙, 因此它的优先级最高, 算法先满足结点 3 的带宽请求, 然后按优先级对其它结点的带宽请求进行分配。该图中, I 链路用于正环通信, II 链路用于反环通信。结点 1 的双向链路用于数据的转发, 结点 2 中一条链路用于数据接收, 而另外一条链路用于数据发送。

3) 每个结点知道全局的路由信息, 预先建立通信链路。因此, 可以保证不出现死锁、活锁以及拥塞现象。

4) 数据环上的结点不需要缓存。

3.3 环路容错机制

在环路中, 每个结点都假设有一个检查数据链路和控制链路失效的机构。如果数据链路失效, 那么通过控制链路向环路上的所有结点广播出错的链路。每个控制结点获取该信息后, 可以避免从失效的链路上发送数据。如果工作的控制链路出现失效, 那么可以通过另外一条备用控制环路向所有结点广播失效信息。图 3 是一条控制环路出现故障之后控制信息倒换到另外一条环路工作的示意图。

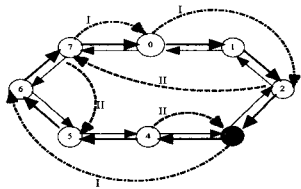


图 2 空分复用图

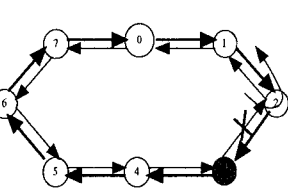


图 3 环路保护机制

4 仿真结果

OPNET 目前已经成为业界领先的网络仿真软件, 广泛应用于网络仿真、网络性能分析、网络管理等各个方面。本文

采用 OPNET 对上述结构进行了仿真。OPNET 采用基于离散事件驱动的仿真机制和基于包的通信机制, 将网络设计模型分为 3 个层次, 即网络模型、节点模型和进程模型, 与真实网络结构相对应。但是针对 NoC 结构, OPNET 并没有相应的模型支持, 因此所有需要的模型都要从进程、节点直至网络层进行建模。为了对本系统进行仿真, 采用 OPNET modeler 对数据节点、控制节点、交换节点以及环网结构从进程模型到网络模型进行了建模。

实验中设置环网节点数分别为 6 和 16 个节点, 每个 IP 的 4 个单工链路以 200M 的速率传输控制、数据信息, 产生的数据包满足泊松分布。在链路工作正常情况下, 使用两条数据链路和一条控制链路进行通信。为方便计算数据时隙和控制时隙, 这里时钟周期按秒来计算。首先由控制链路确定下个数据时隙哪些节点可以发送数据, 数据链路在下一个数据时隙按照控制链路确定的信息进行通信。仿真是在第 10 个时钟周期后开始产生数据流量, 仿真时间持续了 3600 个时钟周期。

图 4(a) 为 6 个节点的环网 NoC 数据产生率分别为 20packet/s, 13packet/s 和 10packet/s 时系统全局平均延时量的对比。随着数据产率的增大, 系统全局平均延时减小。图 4(b) 为数据产生率为 10packet/s 时 6 个节点环网 NoC 中各个控制节点与交换节点之间链路的利用率。图 4(c) 为数据产生率为 10packet/s 时 6 个节点环网 NoC 中各个数据节点与交换节点之间链路的利用率。由于系统采用时分复用、空分复用技术, 控制节点每隔 6 个控制时隙发送一次控制信息, 因此控制链路的利用率较低。而某个数据节点在多个数据时隙内可以发送数据, 多个数据节点在某一数据时隙段内同时发送数据, 因此数据链路的利用率较高。而且, 如果某个数据节点在某个数据时隙可以发送数据, 则该节点的数据链路利用率为 100%。

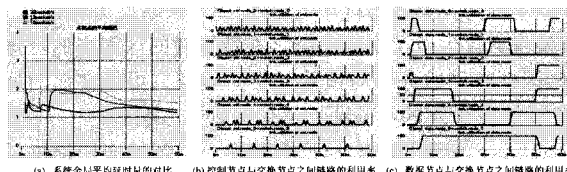


图 4 具有 6 个节点的结构 OPNET 仿真结果

图 5(a) 为 6 个节点和 16 个节点的环网 NoC 数据产生率为 10packet/s 时系统全局平均延时量的对比。随着环网中节点数量的增加, 系统全局平均延时也在增大。图 5(b) 为数据产生率为 10packet/s 时 16 个节点环网 NoC 中各个控制节点与交换节点之间链路的利用率。图 5(c) 为数据产生率为 10packet/s 时 16 个节点环网 NoC 中各个数据节点与交换节点之间链路的利用率。从图 4 和图 5 的仿真结果对比可以看出, 16 节点环网 NoC 相对于 6 节点环网 NoC 链路利用率更高一些, 主要原因是带有 6 个节点的 NoC 不能完全利用系统带宽, 从图中可以看出利用率只有 30% 左右。

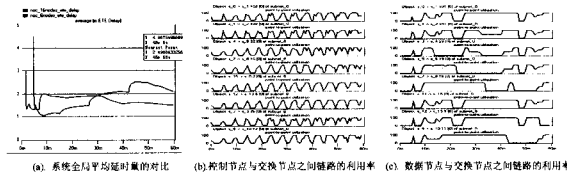


图 5 具有 16 个节点的结构 OPNET 仿真结果

结束语 本文提出了一种分层环形 NoC 拓扑结构,该结构中链路分为两组环网,其中有一组环网为备用环网,用于系统的容错机制。设计了一种时分和空分复用的路由算法、分离控制和数据,去除了数据缓存,消除了链路拥塞,较好地解决了当前 NoC 研究中的几个关键问题。目前正在采用 FPGA 对该结构进行验证,并把该结构用于 TD-SCDMA 的基站的各个单元的通信。

参考文献

[1] Dally W, Towles B. Route packets, not wires: on-chip interconnection networks // Proc. the Design Automation Conference, Las Vegas, NV, 2001; 684-689

[2] Hemani A, Jantsch A, Kumar S, et al. Network on a chip: an architecture for billion transistor era // Proc. IEEE NorChip Conference, 2000; 166-173

[3] Ye T. On-chip multiprocessor communication network design and analysis. PhD Dissertation, Stanford University, 2003

[4] Hu J, Marculescu R. Energy- and performance-aware mapping for regular NoC architectures. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 2005, 24(4); 551-562

[5] Goossens K. Formal methods for networks on chips // Proc. Fifth International Conference on Application of Concurrency to System Design. 2005; 188-189

[6] Hansson A, Goossens K, Rădulescu A. A unified approach to constrained mapping and routing on network-on-chip architectures // Proceedings of the 3rd IEEE/ACM/IFIP international ... 2005

[7] Pirretti M, Link G M, Brooks R R. Fault tolerant algorithms for

network-on-chip interconnect // VLSI, 2004. Proceedings. IEEE Computer Society Annual ... 2004

[8] Bolotin E, Cidon I, Ginosar R, et al. QNoC: QoS architecture and design process for network on chip. Journal of Systems Architecture, 2004

[9] Banerjee N, Vellanki P, Chatha K S. A Power and Performance Model for Network-on-Chip Architectures // Proceedings of DATE, 2004

[10] Jantsch A. The Nostrum Network on Chip. Stockholm; Royal Institute of Technology, 2003

[11] Urbansky R. Transmission system for the synchronous digital hierarchy. US Patent 5,343,476, 1994

[12] Davik F, Yilmaz M, Gjessing S, et al. IEEE 802.17 resilient packet ring tutorial. Communications Magazine, IEEE, 2004

[13] Sigüenza-Tortosa D, Nurmi J, Sigüenza-Tortosa D, et al. Proteo: A New Approach to Network-on-Chip // Proceedings of the IASTED International Conference on Communication Systems and Networks, 2002; 355-357

[14] Samuelsson H, Kumar S. Ring Road Network on Chip Architecture // Proceedings, Norchip Conference, 2004

[15] Guerrier P, Greiner A. A generic architecture for on-chip packet-switched interconnections // Proceedings of DATE 2000, Paris, France, March 2000

[16] Millberg M, Nilsson E, Mid R, et al. Tuaranteed bandwidth using looped containers in temporally disjoint networks within the nostrum network on chip // Proceedings of DATE', Paris, feb, 2004

[17] Dally W J, Towles B. Route Packets, Not Wires: On-Chip Interconnection Networks // Design Automation Conference, 2001

(上接第 60 页)

输出“0”。

以上即为完整的 $ST[t_i, t_j]$ -PETKS-P 原型方案的描述。由于发送方仅生成了指定关键字在特定时间区间内的可搜索加密,因此邮件服务器对该关键字的验证只可能在该时间区间内有效(其最终的可验证性还受接收方影响)。

$ST[t_i, t_j]$ -PETKS-P 原型方案为一类新的应用提供了解决办法,即有效关键字可变的加密邮件。与很多具有可变优先级的计算机体制相类似(例如进程优先级的变化等等),通过 $ST[t_i, t_j]$ -PETKS-P 原型方案可以实现。在邮件服务器中,若邮件等待的时间越长,则该邮件下一次被接收者查看的可能性更高(即具有查看频率更高的关键字)。举例说明如下:

假设关键字 W_1, W_2 的优先级为 $W_1 < W_2$ (优先级越高,接收者查看的频率越高)。发送方采用 $ST[t_i, t_j]$ -PETKS-P 原型方案生成可搜索加密 $S(W_1, [t_i, t_j])$ 和 $S(W_2, [t'_i, t'_j])$, 其中 $t_j < t'_i$, 且令邮件为 $[E_{A_{pub}}[msg], S(W_1, [t_i, t_j]), S(W_2, [t'_i, t'_j])]$ 。显然可以看出,时间 $[t_i, t_j]$ 内由于仅有关键字 W_1 有效,因此该邮件具有较低的优先级。类似地,时间 $[t'_i, t'_j]$ 内关键字 W_2 生效而关键字 W_1 无效,因此该邮件在时间 $[t'_i, t'_j]$ 内具有较高的优先级,从而该邮件在时间 $[t'_i, t'_j]$ 内被接收者查看的可能性增大,即实现了有效关键字可变的加密邮件。

结束语 根据文献[7]提出的 PETKS 原型方案研究了该方案存在的效率问题,即邮件服务器每一次验证都必须进行一次解密操作。本文提出了改进后的方案实例,并在此基础上提出了基于 IBE 方案的 PETKS 原型方案,从而使得该

类原型方案的效率有明显的提高。

另外,本文基于 PETKS 原型方案,提出了两种扩展的原型方案,进一步丰富了 PETKS 的应用。

参考文献

[1] Shamir A. Identity-based cryptosystems and signature schemes [C] // Advances in Cryptology-Proceedings of CRYPTO' 84, LNCS. Vol. 196, Springer-Verlag, 1985; 48-53

[2] Boneh D, Franklin M. Identity-based Encryption from the Weil Pairing [C] // Advances in Cryptology-crypto 2001, LNCS. Vol. 2139, Springer-Verlag, 2001; 231-229

[3] Boneh D, Boyen X. Efficient Selective-ID Identity Based Encryption Without Random Oracles [C] // Advances in Cryptology-EUROCRYPT' 2004, LNCS. Vol. 3027, Springer-Verlag, 2004; 223-238

[4] Boneh D, Boyen X. Secure Identity-based Encryption Without Random Oracles [C] // Advances in Cryptology-crypto 2004, LNCS. Vol. 3152, Springer-verlag, 2004; 443-459

[5] Waters B. Efficient Identity-based Encryption Without Random Oracles [C] // Advances in Cryptology-EUROCRYPT' 2005, LNCS. Vol. 3494, Springer-Verlag, 2005; 114-127

[6] Boneh D, Crescenzo G D, Ostrovsky R, et al. Public Key Encryption with Keyword Search [C] // Advances in Cryptology-EUROCRYPT' 2004, LNCS. Vol. 3027, Springer-Verlag, 2004; 506-522

[7] Abdalla M, Bellare M, Catalano D, et al. Searchable Encryption Revisited: Consistency Properties, Realltion to Anonymous IBE, and Extensions [C] // Advances in Cryptology-Crypto 2005, LNCS. Vol. 321, Springer-Verlag, 2005; 205-222