An Efficient Approach for Managing Power Consumption Hotspots Distribution on 3D FPGAs

Kostas Siozios and Dimitrios Soudris

VLSI Design and Testing Center, Department of Electrical and Computer Engineering
Democritus University of Thrace, 67100, Xanthi, Greece
{ksiop, dsoudris}@ee.duth.gr

Abstract. Using new silicon technologies, increasing logic densities and clock frequencies on FPGAs lead to rapid elevation in power density. Since the power consumption is a critical challenge for application implementation, a novel power-aware partitioning, placement and routing (P&R) algorithm targeting to 3D FPGAs, is introduced. The proposed methodology achieves to redistribute the switched capacitance over the hardware resources in a rather "balanced" profile, reducing among others the maximal on-chip temperatures. Due to the relation between switched capacitance and power consumption, the proposed P&R algorithm can be considered as a power management approach. This algorithm is realized as part of 3DPRO tool. Comparing to alternative P&R solutions, we eliminate the area on hotspots about 68%, while we achieve savings in delay and energy consumption about 9% and 11% in average, respectively.

1 Introduction

For decades, semiconductor manufacturers have been shrinking transistor size in ICs to achieve the yearly increases in speed and performance described by Moore's Law, which exists only because the RC delay was negligible in comparison with signal propagation delay. For submicron technology, however, the RC delay becomes a dominant factor. This has generated many discussions concerning the end of device scaling as we know it, and has hastened the search for solutions beyond the perceived limits of current 2D devices.

One emerging solution to this problem is the 3D integration, which replaces a large number of long interconnects needed in 2D structures with shorter ones. Such architectures mitigate many of the limitations that the 2D devices exhibit. Among others, are: (*i*) higher logic density in the same foot print area, (*ii*) shorter interconnections among the logic blocks, (*iii*) reduced signal propagation delay, (*iv*) greater versatility and resource utilization, and (*v*) lower power consumption.

One of the most critical challenges for efficient application implementation in 3D FPGAs is the power management, and hence the thermal problem, which has already been studied for 2D architectures [3, 6, 7, 8]. This problem is exacerbated in the 3D devices for two reasons: (*i*) the vertically stacked layers cause a rapid increase of power density [9] and (*ii*) the thermal conductivity of the dielectric layers inserted between device layers for insulation is very low compared to silicon and metal [15].

Moreover, an obvious consequence of this trend is the increased power consumption per area unit. In recent years, power density in 2D FPGAs has doubled every three years [1], and this rate is expected to increase as feature sizes, frequencies and technologies scale faster than operating voltages. As the power density will continue increasing in future technologies (according to "*A-power*" law [3]), the power consumption is regarded as a limiting factor to the increasing scales of integration predicted by Moore's law [1].

Eliminating and managing power consumption requires appropriate algorithm support. Realizing applications on 2D FPGAs is a well studied problem; however, there are only a few solutions for 3D architectures [4, 13, 14].

In [13] a P&R approach for 3D ICs is presented, having as criterion to minimize the total wire length, the applications delay, and the on-chip temperature. Even though this framework supports reconfigurable architectures, however, the thermal feature is available only for the ASIC designs.

A similar approach for 3D FPGAs is shown in [14], where the P&R algorithm optimizes the energy consumption and the thermal profile of a standard-cell circuit under the supplied timing constraint. This algorithm focuses on the interconnect-related components of energy consumption that can be affected by placement based optimization. Unfortunately, the software implementation is not publically available, in order to evaluate this approach against to our proposed solution.

In [4] a thermal-driven 3D floor-planning algorithm is presented. This algorithm trade-offs between runtime and quality. The goal is to reduce the total wire-length, as well as the maximum on-chip temperature, compared to a conventional (i.e. non-thermal-driven) 3D floor-planning approach.

In [8] a P&R algorithm and its software implementation targeting to explore alternative interconnection schemes for 3D FPGAs are introduced. The employed cost functions pay effort to minimize the application delay, the power/energy consumption, as well as the total wire length, ignoring about their distribution. This tool is part from an open-source CAD framework, named *3D MEANDER*, for mapping applications on 3D FPGAs.

All these approaches realize digital applications on 3D devices having as goal to minimize the total power/energy consumption of the design, ignoring the spatial distribution of its sources. Moreover, none of them takes into consideration during the P&R the spatial distribution of the parameters that affect power/energy consumption (i.e. switched capacitance). This results to higher power/energy consumption, and consequently temperature, variations across the 3D device. Among others, this non-uniformity in power consumption leads to increased cooling costs, as the IC packaging has to be designed for the worst case scenario.

1.1 Problem Formulation

Power consumption of FPGAs is generally grouped into three categories: Dynamic power, static power, and interface (I/O) power. These components are governed by the silicon process technology and traditionally have maintained constant percentages of the device's total power. The dynamic part of power consumption (formulated in the following equation), which occurs due to signal transition as the load capacitance is charged (or discharged) is the dominant component of the total power consumption. In this equation, f represents the clock frequency of the signal, V_{dd} is the supply voltage, while Cap_i and $Activity_i$ are the capacitance and switching activity, respectively, of hardware element i.

$$P_{switching} = 0.5 \cdot f \cdot V_{dd}^2 \cdot \sum_{i=1}^{Nets} \{Cap_i \cdot Activity_i\} \quad (1)$$

When a lower bound on the supply voltage is set by external constraints (as often happens in real-world designs), or when the performance degradation due to lowering of the supply voltage is intolerable, then the only means of reducing the power consumption is by lowering the effective capacitance and the switching activity (*i.e.*, switched capacitance). Throughout the paper, we discuss algorithms that control the spatial distribution of this product ($Cap_i \cdot Activity_i$), leading to a more "*uniform*" distribution of it across the FPGA device.

Definition: Application Graph
We consider as application graph a directed graph $AppG(L, N)$, where each vertex $l_i \in L$ represents a logic function of the application, while the directed edge $n_{i,j} \in N$ corresponds to the communication between the logic functions l_i and l_j. The weight of the edge $n_{i,j}$ denoted as $communication\_weight_{i,j}$, represents the communication load/bandwidth from vertex l_i to l_j.

Definition: Platform Graph
We consider as platform graph a directed graph $PlatG(C, W)$ where each vertex $c_i \in C$ represents an element of the target architecture (*e.g.*, logic block, processor, memory, etc.), while the directed edge $w_{i,j} \in W$ represents a communication path between the hardware elements c_i and c_j. The weight of the edge $w_{i,j}$, denoted as $interconnection\_weight_{i,j}$, denotes the fabricated interconnection hardware resources among these logic blocks.

3D Placement and Routing Problem
Given the architecture graph $PlatG(C, W)$ consisted by a set of V ($V = i \times j \times k$) slices (S), where $S = \{S_1(x_1, y_1, z_1), \cdots, S_v(x_i, y_j, z_k)\}$ find a placement ($Place: L \rightarrow C$) and a routing ($Route: N \rightarrow W$) of the application graph on the available hardware resources (platform graph), in order each logic function (L) and the appropriate communication (N) to occupy uniquely a logic resource (C) and the available routing fabric (W), respectively. Each P&R solution is accepted if the following conditions are satisfied:

(i) $S_i \cap S_j = 0$ for all the $i, j \in PlatG$
(ii) The interconnection of each layer is accomplished with the minimum routing resources
(iii) Employ the minimum number of vertical links (*i.e.* vias)
(iv) Meeting the timing/power/area constraints of the application
(v) Distribute uniformly the switched capacitance over the 3D device

The proposed power-aware P&R solution was evaluated against to the conventional (i.e. non power-aware) P&R mapping with the usage of the 20 biggest MCNC benchmarks [5]. During this experimental setup, the two P&R algorithms are applied to identical (i.e. with same amount of logic resources and interconnection fabric) 3D FPGAs. The results show significant reduction of power consumption on the hotspot regions (around 68%). In addition to that, we achieve to reduce the applications delay and its energy consumption about 9% and 11% (in average), respectively.

The rest paper is organized as follows. In Section 2, the algorithmic steps of the proposed P&R approach are introduced. Section 3 presents the comparison results with a non power-aware mapping, while conclusions are summarized in Section 4.

2 Proposed Placement and Routing Algorithm

Figure 1 shows the tool flow, named *3D MEANDER*, for realizing applications on 3D FPGAs. This flow adopts some existing CAD tools from the 2D toolset [8], which do not need to be aware of the 3D FPGA topology (*i.e.*, technology platform independent). To the best of our knowledge, this toolset is the first complete framework in academia for mapping applications on 3D reconfigurable devices starting from hardware description language up to configuration file generation.

The proposed power-aware algorithm is implemented within the *3DPRO* tool. In order to measure the delay we employ the Elmore delay model [12], while regarding the power/energy estimations we propose an enhanced version of models introduced in [2]. These models are integrated in the *3DPower* tool.

As our proposed P&R approach is power-aware, it poses new challenges to application implementation on 3D devices. Detail description of each algorithmic step (*i.e.*, partitioning, placement, routing) will be given in the upcoming sections.

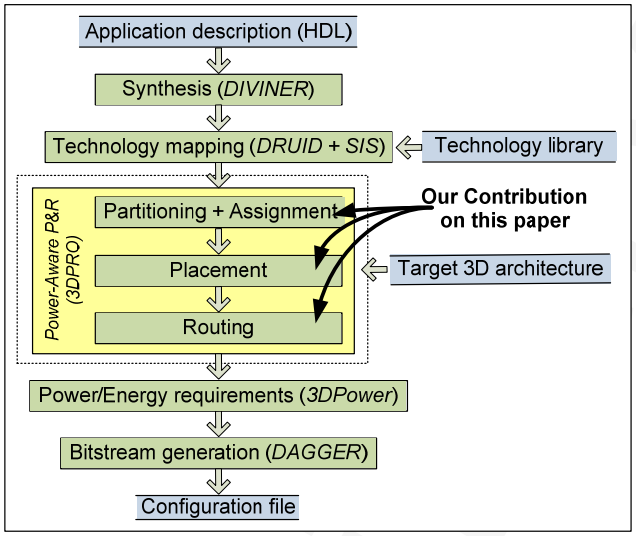


Figure 1: The 3D MEANDER Framework

2.1 Application Partitioning and Layer Assignment

The first step of the proposed power-aware P&R algorithm deals with the application partitioning into a number (equals to the device layers) of balanced sections. The employed partitioning algorithm is based on [11], as it tries to minimize the interlayer communication, however its cost function has appropriately extended to spread as much as possible the spatial distribution of applications switched capacitance across them, without increasing the total power/energy consumption, the applications delay or silicon area requirements.

This procedure is done by recursive bi-partitioning of the application graph $AppG$, such that to minimize the value of the employed cost function. Since the net length is tightly firmed to its resistance and capacitance values, we can manage the power consumption sources by weighting each net according to its switched capacitance.

We associate the switched capacitance criticality of a logic element as weight to the corresponding vertex in the hypergraph, while the timing criticality is shown as weight to the corresponding hyperedge. These weights encourage the partitioning algorithm to split the application in a way that balances both of them. The criticalities of the graph (i.e. weights of vertexes and hyperedges) are updated at each partitioning level. The partitioning process stops when both the switched capacitance distribution and the timing constraints are met.

Next, the algorithm assigns these ports on the device layers by taking into consideration a number of design parameters. More specifically, the algorithm tries not to assign layers that consume high power close to each other, or on the middle of the 3D stack, as it is more difficult to dissipate heat. This task is accomplished in conjunction to the available interlayer communication (i.e. vias connections) or other design constraints (i.e. maximum acceptance delay, power/energy consumption, etc).

2.2 Application Placement

After the partitioning step, the placement algorithm assigns the application's logic functions (L) to available hardware modules (C). As the majority of applications realized onto FPGAs utilize only a subset of the available hardware resources, this non-uniformity leads to high variation of power consumption across the device [8]. This problem gets even worst in 3D devices due to high power/temperature variation among layers.

The proposed power-aware placement algorithm tries to place the logic functions (L) in a way that minimizes the maximal switched capacitance values (referred as *hotspots*), as well as to distribute it across the whole 3D FPGA. As the switching activity depends on the functionality implemented inside the logic modules, while the capacitance is proportional to the interconnection length and the number of hardware modules that form each network, the proposed algorithm handles in an efficient way their product (i.e. switched capacitance). More specifically, by placing on adjacent spatial locations logic functions connected through nets with high switching activity, these nets probably will be shorter (exhibit smaller capacitance), leading to reduced power consumption. Unfortunately, it is not always possible to place close all these blocks, as this might lead to increased application delay (*i.e.* delay of the slowest path). The employed cost function that guides our proposed placer follows:

$$\Delta Cost = \left\{\alpha \times \frac{\Delta Wire_{cost}}{Previous\ Wire_{cost}} + (1-\alpha) \times \frac{\Delta Time_{cost}}{Previous\ Time_{cost}}\right\} \times \frac{\Delta Activity_{cost}}{Previous\ Activity_{cost}} \qquad (2)$$

where $Wire_{cost} = \sum_{i=1}^{Nets}\{q(i) \times [bb_x(i) + bb_y(i) + bb_z(i)]\}$, $Time_{cost} = \sum_{i=1}^{Nets}\{delay(i) \times critical(i)^{\beta}\}$ and $Activity_{cost} = \sum_{i=1}^{Nets}\{activity(i)\}$.

The factor α of cost function balances the effort for reducing either the total wire length or the delay. However, in both cases, the algorithm tries to reduce the switching activity. The $bb_x(i)$, $bb_y(i)$ and $bb_z(i)$ parameters denote the dimensions of the 3D bounding box for network i, while the $q(i)$ is a scaling factor of the bounding box, used to make more accurate estimations about the wire-length for nets with more than 3 terminals [10]. The $delay(i)$ denotes the delay between a source-sink path of a network, the factor β is a constant, while the $critical(i)$ gives the importance, in terms of how close to the critical path, is the network i. Finally, the $activity(i)$ represents the switching activity value for the network i. In order to calculate this parameter, the transition density for all the hardware elements of network i has to be summarized.

2.3. Application Routing

By defining the placement on the 3D FPGA, the routing algorithm forms the appropriate connections among the utilized logic blocks (C) through the available interconnection fabric (W). The proposed routing algorithm is based on Pathfinder negotiated congestion. During the first iterations, a number of networks are allowed to share the same routing fabric. However, as the number of iterations increase, this is gradually prohibited, until to the final routing where each network uses dedicated routing fabric. Such a router finds the narrowest horizontal and vertical channel widths for which the application is fully routable.

As the vertical interconnections are limited, compared to horizontal tracks, the routing algorithm in order to discourage the router to form unnecessary bends between horizontal and vertical wires sets their weight to a higher value. Also, this penalty forces the router not to connect logic blocks placed on one layer by using interconnection fabric from different layers.

Our proposed routing algorithm tries to spread the switched capacitance in a more uniform manner across the 3D device, while it achieves the timing and total power/energy constraints. Due

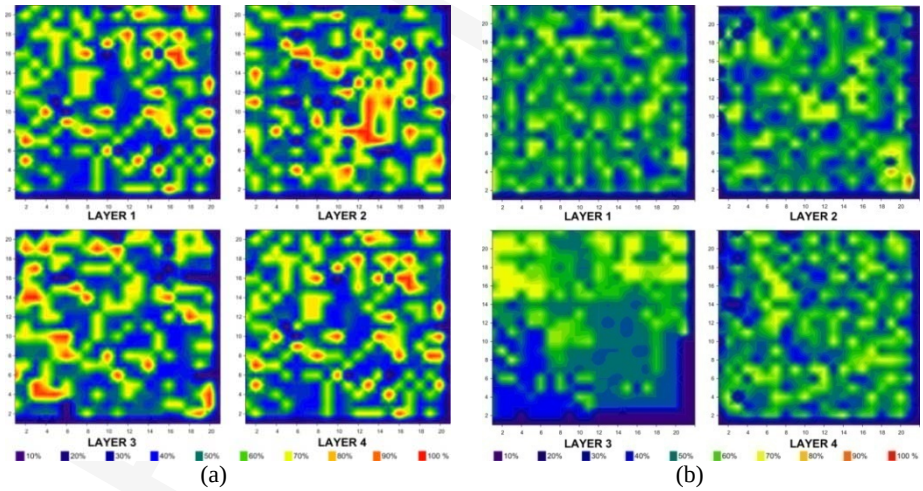
to this, it avoids (whenever this is feasible from application constraints) to form routing paths that cross regions with high power consumption (e.g. high switched capacitance).

3. Experimental Results

We implement the proposed power-aware P&R algorithm in C++, as part to an existing open-source tool for 3D FPGAs, named *3DPRO*. For evaluation purposes, the 20 biggest MCNC benchmarks are mapped on a 3D FPGA which architecture is inspired by the one that proposed in [8]. Such a 3D device is constructed by stacking a number of identical 2D FPGAs on individual functional layers, providing appropriate communication among them by interlayer vias. These connections are realized inside vertically adjacent 3D Switch Boxes. The employed 3D architecture has a vias distribution with smaller fabrication costs compared to conventional 3D FPGAs, without any degradation in application performance, or increment of total power/energy consumption. The features of this architecture are summarized as follows:

(i) The total number of functional layers is equal to four.
(ii) The percentage of vertical interconnections (i.e. vias) per layer is 30%.
(iii) The spatial locations for vertical interconnections per layer remain invariant.
(iv) The vertical interconnection was modelled based on the approach shown in [15].
(v) There are 4 bit connections between layers for each 3D SB.
(vi) All the layers have identical resources.
(vii) The applications are mapped onto the smallest 3D FPGA that fits.

Figure 2 shows the variation of switched capacitance across the 3D FPGA device layers. In order to derive these graphs, the *alu4* (one of the 20 biggest MCNC benchmarks), consisted by 1522 4-input LUTs, is employed. More specifically, Figure 2(a) shows the variation of switched capacitance for the 3D FPGA with a conventional P&R, while Figure 2(b) gives the variation resulted with the proposed power-aware algorithm. From these graphs, it is evident that the conventional P&R exhibits both higher maximal values of switched capacitance, as well as higher variation. Consequently, the gradient of on-chip temperature is higher, as compared to the proposed approach.



(a) (b)

Figure 2: Variation of switched capacitance across the 3D FPGA device layers (a) without and (b) with the proposed power-aware P&R

Apart from the switched capacitance distribution, the proposed algorithm tries not to increase either the application's delay or its total power/energy consumption. Table 1 summarizes the evaluation results of applying the proposed power management strategy on the 20 biggest MCNC benchmarks. More specifically, we perform comparisons in terms of the application delay, the energy consumption, as well as the percentage of silicon area that belongs to *hotspot*. By the term *hotspot* we refer device area that consumes more than 70% of the maximum power consumption.

Table 1 proves that the proposed power-aware algorithm achieves to reduce the area percentage that operates under high power (*hotspot* regions). More specifically, the column marked as "*Area on hotspot*" quantifies the main goal of the developed research. Based on the results provided in Table 1, we conclude that the proposed P&R algorithm reduces the percentage of area that operates under high power. In addition to that, the derived mapping with the proposed power-aware P&R achieves to speed up the operation frequency (i.e. reduce the applications delay) about 9%, while it leads to energy savings about 11%, in average. These gains in delay and energy consumption occur due to the different resistance and/or capacitance values appeared at the interconnection fabric of the 3D FPGA device, since logic functions are partitioned, placed and routed with different cost functions.

Table 1. Comparison in terms of delay, energy consumption and device area marked as *hotspot* between a conventional and the proposed (power-aware) P&R on identical 3D FPGAs

Benchmark	Conventional (non power-aware) P&R			Proposed (power-aware) P&R		
	Delay (nsec)	Energy (nJ)	Area on hotspot(%)	Delay (nsec)	Energy (nJ)	Area on hotspot(%)
alu4	3.82	6.18	26%	3.48	5.26	5%
apex2	4.37	8.30	30%	3.84	7.14	5%
apex4	3.28	4.66	16%	2.85	4.24	6%
bigkey	1.64	10.0	24%	1.46	9.43	6%
clma	5.34	44.5	18%	4.76	41.8	9%
des	3.28	13.0	30%	2.82	12.2	10%
diffeq	4.21	11.9	16%	3.58	10.5	9%
dsip	1.64	7.15	30%	1.46	6.22	9%
elliptic	3.84	13.3	30%	3.69	12.0	5%
ex1010	4.37	12.7	19%	3.71	11.7	5%
ex5p	2.45	4.14	20%	2.23	3.72	8%
frisk	3.01	26.4	15%	2.89	22.4	7%
misex3	3.82	5.64	23%	3.52	5.19	8%
pdc	4.91	19.2	23%	4.62	16.3	9%
s298	8.19	10.2	23%	7.37	8.86	10%
s38417	7.45	43.1	20%	6.71	38.8	5%
s38584	6.97	31.1	17%	6.55	28.3	9%
seq	3.82	7.15	30%	3.67	6.44	6%
spla	4.37	12.6	23%	4.19	11.3	9%
tseng	4.11	19.3	25%	3.58	16.6	8%
Average:	**4.24**	**15.5**	**22.9%**	**3.85**	**13.9**	**7.4%**
Ratio:	**1.00**	**1.00**	**1.00**	**0.91**	**0.89**	**0.32**

The gains of the proposed power-aware P&R approach can be summarized as follows: (*i*) spreads the power sources across the whole 3D FPGA in a way that it is more easy to dissipate heat, (*ii*) reduces the peak values of power leading to cheaper fabrication cost for cooling, (*iii*) reduces the total energy consumption increasing among other the battery life and system reliability, (*iv*) increases the maximum operation frequency, and (*v*) reduces the percentage of silicon area that consumes high power. Based on these, our proposed power-aware P&R algorithm, as well as its software implementation through the *EX-VPR* tool, achieves better application mapping onto 3D reconfigurable architectures, while it can be though as a power management approach.

4. Conclusions

An efficient approach for managing power consumption hotspots distribution, as well as its software implementation, targeting 3D FPGAs was presented. This approach can be though as a power management strategy, as it achieves to re-distribute the power budget over identical hardware resources in a way that the produced heat is easily to be dissipated. Also, there is no impact either on the application delay (reducing by 9%) or on the total energy requirements (savings about 11%). More specifically, the proposed P&R algorithms reduces about 68%, in average, the percentage of device area that operates under high power by controlling appropriately the switched capacitance.

Acknowledgement

This work was supported by the project PENED 03ED593 which is funded by the GSRT of Ministry of Development.

References

1. "International Technology Roadmap for Semiconductors", *electronic document available at http://www.intel.com/technology/silicon/itroadmap.htm*
2. K. Poon,et.al.,"A Flexible Power Model for FPGAs", *Proc. of 12th FPL*, pp.312–321, 2002.
3. T. Sakurai and A. R. Newton, "Alpha-power law mosfet model and its applications to cmos inverter delay and other formulas", *IEEE JSSC*, April 1990.
4. J.Cong,et.al.,"A Thermal-Driven Floorplanning Algorithm for 3D ICs", *Proc. of ICCAD*,2004.
5. S. Yang, "Logic Synthesis and Optimization Benchmarks, Version 3.0", *Tech. Report, Microelectronics Centre of North Carolina*, 1991.
6. A. Telikepalli, "Designing for Power Budgets and Effective Thermal Management," *In Xcell Journal*, Issue 56, 2006.
7. "Thermal Management for 90-nm FPGAs", *Application Note 358*, Altera Corporation.
8. K. Siozios,et.al., "Exploring Alternative 3D FPGA Architectures: Design Methodology and CAD Tool Support", *17th Int. Conf. on Field Programmable Logic and Applications*, 2007
9. T.Y. Chiang, et.al., "Thermal Analysis of Heterogeneous 3D ICs with Various Integration Scenarios", *Technical Dig. IEDM*, 2001, pp.681-684.
10. V. Betz, J. Rose, and A. Marquardt, "Architecture and CAD for Deep-Submicron FPGAs", *Kluwer Academic Publishers*, Feb. 1999.
11. N. Selvakkumaran and G. Karypis, "Multi-Objective Hypergraph Partitioning Algorithms for Cut and Maximum Subdomain Degree Minimization", *Proc. of ICCAD*, pp. 726, 2003.
12. T. Okamoto and J. Cong, "Buffered Steiner Tree Construction with Wire Sizing for Interconnect Layout Optimization", *Proc. of ICCAD*, pp. 44-49, 1996.
13. Cristinel Ababei, et.al., "Placement and Routing in 3D Integrated Circuits", *IEEE Design & Test of Computers*, Vol. 22, No. 6, pp. 520-531, 2005.
14. S. Das, et.al., "Timing, Energy, and Thermal Performance of Three Dimensional Integrated Circuits", *Proc. of GLSVLSI*, pp. 338-343, 2004.
15. A. Rahman, J. Trezza, B. New, and S. Trimberger, "*Die Stacking Technology for Terabit Chip-to-Chip Communications*," Proceedings of the IEEE Custom Integrated Circuits Conference, pp. 587–590, 2006