Speedup Requirements for Output Queuing Emulation with a Sliding-Window Parallel Packet Switch

Chia-Lung Liu[1], Woei Lin[1], and Chin-Chi Wu[2]

[1] Department of Computer Science, National Chung-Hsing University,
250, Kuo Kuang Road, Taichung, Taiwan
{s9056005, wlin}@cs.nchu.edu.tw
[2] Nan Kai Institute of Technology
wcc007@nkc.edu.tw

Abstract. This investigation uses an approximate Markov chain to determine whether a sliding window (SW) parallel packet switch (PPS), only operating more slowly than the external line speed, can emulate a first-come first-served (FCFS) output-queued (OQ) packet switch. A new SW packet switching scheme for PPS, which is called SW-PPS, was presented in the authors' earlier study [1]. The PPS class is characterized by deployment of distributed center-stage switch planes with memory buffers that run slower than the external line speed. Given identical Bernoulli and Bursty data traffic, the proposed SW-PPS provided substantially outperformed typical PPS, in which the dispatch algorithm applies a round-robin method (RR) [1]. This study develops a presented Markov chain model that successfully exhibits throughput, cell delay and cell drop rate. A simulation reveals that the chains are accurate for reasonable network loads. Major findings concerning the presented model are that: (1) the throughput and cell drop rates of a SW-PPS can theoretically emulate those of a FCFS-OQ packet switch when each slower packet switch operates at a rate of around R/K (Eq. 19); and, (2) this investigation also proves the theoretical possibility that the cell delay of a SW-PPS can emulate that of an FCFS-OQ packet switch, when each slower packet switch operates at a rate of about (R/cell delay of FCFS-OQ switch) (Eq. 20).

1 Introduction

For efficient data deliver in high-capacity switching systems, the PPS is a good choice in a high-speed communication [1]–[3]. However, traditional PPS which uses a round-robin method cannot effectively use the memory space of center-stage switches, and so requires much memory [3]. Our previous study describes a novel SW packet switching method for PPS, called SW-PPS [1]. This novel switching scheme overcomes the shortcomings of traditional PPS, and uses memory space more effectively than traditional PPS. Accordingly, previous experimental results demonstrate that SW-PPS outperformed traditional PPS [1].

Work conserving [2] refers to a situation in which a switch's output port is working whenever there is a cell in the switch for it. FCFS-OQ switches are work conserving. However, FCFS-OQ switches require buffer memories running at N times

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part IV, LNCS 3994, pp. 49 – 56, 2006.

external line rates. A necessary condition for a switch to emulate the performance of FCFS-OQ switch is that it be work conserving. Under identical inputs, if switch A emulates switch B, the performance of switch A equals or transcends that of the switch B. Therefore, this study reveals that the throughput and drop rate of the SW-PPS can emulate those of a FCFS-OQ switch with $S \geq 1$ (Eq. 19), and that cell delay of the SW-PPS can emulate that of a FCFS-OQ switch with $S \geq K/\text{delay}$ of FCFS-OQ switch (Eq. 20), where S is the speedup of the internal line speed (R/K).

2 SW-PPS Architecture

The SW-PPS is divided into the following independent stages: 1) the self-routing parameter assignment circuit; 2) the demultiplexers; 3) the slower speed center-stage packet switches; and, 4) the multiplexers (shown in figure 1). The destined output port of the incoming cell, extracted by applying header processing circuits, is indicated by *d*. The incoming cell's destination address *d* is delivered to a self-routing parameter assignment circuit. In processing incoming cells, the self-routing parameter assignment circuit employs the output port *d* and a parameter assignment algorithm to create an additional group of self-routing parameters (*i, j, and d*). These self-routing parameters (*i, j, and d*) are attached to incoming cells as a self-routing tags. Incoming cells then use the attached tags to navigate through the demultiplexers and center-stage switches. Parameters (*i, j, and d*) are defined as follows: the variable *i* in parameters informs the center-stage switch where the cell will be stored; variable *d* indicates which memory module in the *i*th center-stage switch the cell will be stored in; and, variable *j* designates the memory location in the *d*th memory module where the cell will be stored. During the cell WRITE cycle for an incoming cell, the cell is written to *j*th memory location in a given memory module *d* and a given center-stage switch *i*. During the cell READ cycle, cells are sent to multiplexers according to the location of the SW. The number of center-stage switches refers to the reference [2].

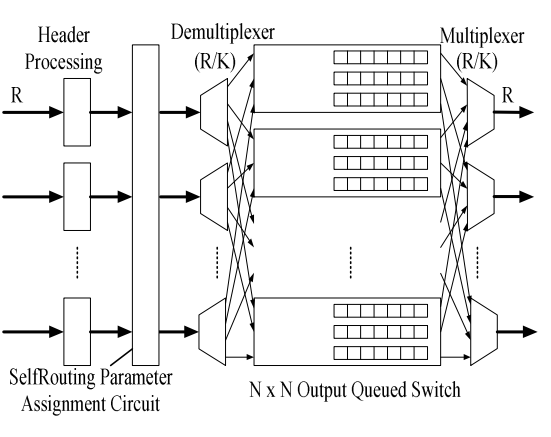

Fig. 1. Architecture of the SW-PPS

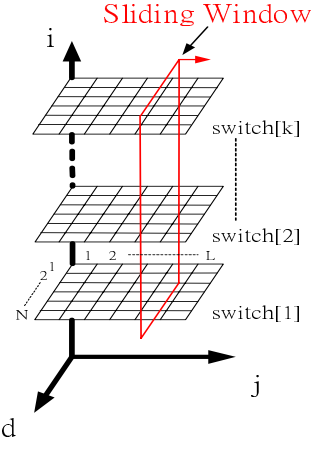

Fig. 2. A 3-D representation of memory space

3 SW Switching Scheme

According to the SW-PPS switching schedule, the overall memory space, including all cell memory locations in all of the center-stage switches, is represented as a three-dimensional (3-D) memory space (*i, j, and d*) (shown in figure 2). The memory locations in the global memory space are represented by a 3-D coordinate system (*i, j, and d*), where the *i*th coordinate represents the center-stage switch; *i*= [1…K], where K is the number of center-stage switches. The *d*th coordinate indicates the memory module; *d*= [1…N], where N is the size of PPS (or the size of the center-stage switch); *j*th coordinate designates the memory location in the memory module; *j*= [1…L], where L is queue size. In other words, L represents the number of memory locations in each memory module. The SW is regarded as a pointer to a group of cells in the 3-D memory space (shown in figure 2). The SW advances one step during each switch cycle. The location of the SW in the global memory space is recorded by one variable: SW.*j*.

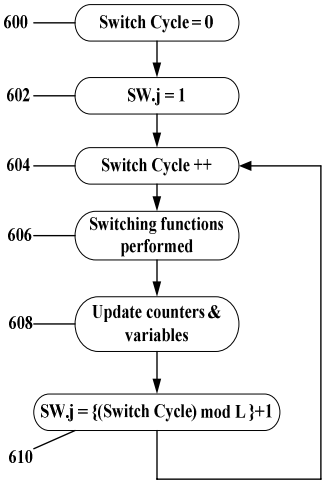

Fig. 3. Flowchart depicting traversal of SW

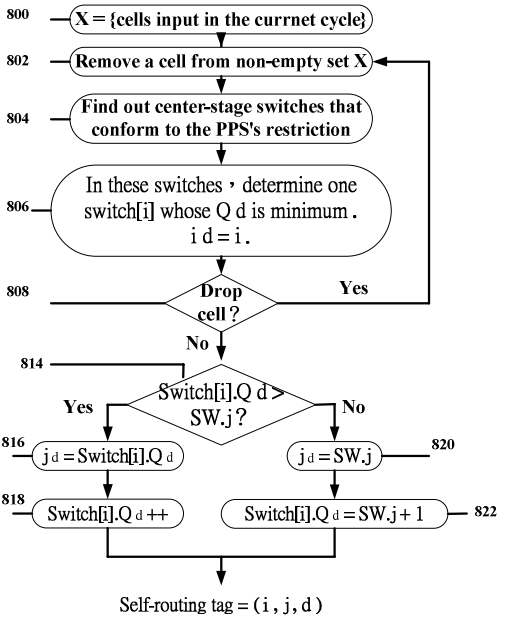

Fig. 4. Assignment process for self-routing parameters (i, j, and d)

The flowchart in figure 3 shows SW traversal across the entire 3-D memory space and its connection to the switch cycle and switching operation. After switching functions in step 606, counters and variables are updated in 608 to account for changes. The SW is then advanced to next location in step 610.

Assigning self-routing parameters (*i, j, and d*) to incoming cells is computed by the parameter assignment circuit. An additional routing tag carrying the self-routing parameters (*i, j, and d*) is attached to incoming cells. Before exploring the process the

self-routing parameters (*i, j, and d*), the PPS's restrictions first be understood [1][2]. Determination of self-routing parameters (*i, j, and d*) by the assignment circuit to an incoming cell is shown by the flowchart in figure 4. The symbols used therein are described as follows, (1) *d* is the cell's output-port destination (2) (i_d, *and* j_d) are the parameters (*i, and j*) of the incoming cell sent to output port *d* (3) *m* is the number of the fanout, when it is in the multicast traffic (4) $Switch[i].Q_d$ is the memory location that is written next time inside the *d*th memory module and *i*th center-stage switch for the cells destined to output port *d*. (5) *X* is a set of cells input during a given external cycle, $0 \le |X| \le N$, where N is the number of input ports. According to step 808 (shown in Fig. 4), if the number of cells in the *d*th queue of the *switch[i]* is greater than L, the cell destined to d is dropped.

4 Analysis for FCFS-OQ Switch

The performance of a FCFS-OQ switch is first analyzed. The following assumptions are made: (1) Cells arrive according to a Bernoulli process. (2)Traffic load is uniformly distributed. (3) The length of the cell is constant. (4) Each queue has a finite capacity.

The following is a list of notations which are used in the development of the performance model and its subsequent analysis. (1) N: Size of the FCFS-OQ switch. (2) ρ: Input load. (3) L: Length of buffer. (4) P_j (t): Probability that j cells are stored in a buffer at network cycle t. (5) P_{drop} (t): Probability that the buffer overflows at network cycle t. (6) g_i: Probability that i cells arrive at same output queue. (7) r: Probability that a cell in a queue successfully moves to the output port. (8) $\bar{r}$: $\bar{r} = 1 - r$.

So we assume the probability r = 1, there is no head-of-line blocking (HOL) problem in FCFS-OQ switch [5]. The FCFS-OQ switch load= ρ and r= 1 obtains g_i and following equations.

$$g_i = C_i^N * (\frac{\rho}{N})^i * (1 - \frac{\rho}{N})^{N-i}, \qquad 0 \le i \le N \cdot \tag{1}$$

$$P_0(t+1) = P_0(t) * (g_0 + g_1 * (r)) + P_1(t) * (g_0) * (r) \cdot \tag{2}$$

$$P_j(t+1) = \sum_{n=0}^{j} P_n(t) * (g_{j-n}) * (\bar{r}) + \sum_{n=0}^{j+1} P_n(t) * (g_{j+1-n}) * (r), \ 1 \le j \le L-1 \cdot \tag{3}$$

$$P_L(t+1) = \sum_{n=0}^{L} P_n(t) * (g_{L-n}) * (\bar{r}) + \sum_{n=0}^{L} P_n(t) * (g_{L+1-n}) * (r) + P_{drop}(t) \cdot \tag{4}$$

$$P_{drop}(t+1) = \sum_{n=0}^{L} \sum_{i=L+1-n}^{N} P_n(t) * (g_i) * (\bar{r}) + \sum_{n=0}^{L} \sum_{i=L+2-n}^{N} P_n(t) * (g_i) * (r) \cdot \tag{5}$$

The three primary performance measures are given as follows.

$$FCFS\_OQ\_Switch\_Drop \quad Rate(\rho, S, t, N, K, L) = P_{drop}(t) \cdot \tag{6}$$

$$FCFS\_OQ\_Switch\_Throughput \ \ (\rho, S, t, N, K, L) = (\rho - P_{drop}(t)) \cdot \tag{7}$$

$$FCFS\_OQ\_Switch\_Delay(\rho, S, t, N, K, L) = \frac{[\sum_{i=1}^{L}(i)*(P_i(t))] + L*(P_{drop}(t))}{1-P_0(t)} \cdot \quad (8)$$

5 SW-PPS Analysis

The SW-PPS was further simplified to an output buffer that is described by a Markov chain model. Finally, four equations of performance evaluations are derived. The assumptions made herein are the same as those in section 4 with one additional assumption: Low speed center-stage switches are OQ switches.

For clarity, we list notations that are employed in the development of the performance model and analysis. (1) N: Size of the SW-PPS. (2) K: Number of center-stage switches. (3) ρ: Input load. (4) L_{sw}: Size of buffer. (5) P_{sw_j} (t): Probability that j cells are stored in a buffer at network cycle t. (6) P_{sw_drop} (t): Probability that buffer overflows at network cycle t. (7) g_{sw_i}: Probability that i cells arrive at the same output buffer. (8) r_{sw}: Probability that a cell in a buffer successfully moves to the multiplexer. (9) $\bar{r}_{sw}$: $\overline{r_{sw}} = 1 - r_{sw}$. (10) S: The speedup of the internal link.

Because (1) External load= ρ, (2) number of center-stage switches= K, (3) SW scheme will choose one of the switches that Q_d is minimum (shown in steps 804 and 806 in the figure 4), we obtained internal load of the SW-PPS (ρ'), g_{sw_i} and r_{sw}.

$$\rho' = \frac{\rho}{C_1^{K(1-\rho)+1}} \cdot \quad (9)$$

$$g_{sw\_i} = C_i^N * (\frac{\rho'}{N} * \frac{1}{S})^i * (1 - \frac{\rho'}{N} * \frac{1}{S})^{N-i}, \qquad 0 \le i \le N \cdot \quad (10)$$

We assume the probability r_{sw} = 1, because there is no head-of-line blocking (HOL) problem in center-stage switches that are OQ switches.

The resulting equations are similar to those for a Markov chain of the FCFS-OQ switch (section 4), because the center-stage switches use FCFS-OQ switches. We have following equations.

$$P_{sw\_0}(t+1) = P_{sw\_0}(t)*(g_{sw\_0} + g_{sw\_1} * r_{sw}) + P_{sw\_1}(t)*(g_{sw\_0})*(r_{sw}) \cdot \quad (11)$$

$$P_{sw\_j}(t+1) = \sum_{n=0}^{j} P_{sw\_n}(t)*(g_{sw\_j-n})*(\bar{r}_{sw}) + \sum_{n=0}^{j+1} P_{sw\_n}(t)*(g_{sw\_j+1-n})*(r_{sw}), \ 1 \le j \le L_{sw}-1 \cdot \quad (12)$$

$$P_{sw\_L_{sw}}(t+1) = \sum_{n=0}^{L_{sw}} P_{sw\_n}(t)*(g_{sw\_L_{sw}-n})*(\overline{r_{sw}}) + \sum_{n=0}^{L_{sw}} P_{sw\_n}(t)*(g_{sw\_L_{sw}+1-n})*(r_{sw}) + P_{sw\_drop}(t) \quad (13)$$

$$P_{sw\_drop}(t+1) = \sum_{n=0}^{L_{sw}} \sum_{i=L_{sw}+1-n}^{N} P_{sw\_n}(t)*(g_{sw\_i})*(\overline{r_{sw}}) + \sum_{n=0}^{L_{sw}} \sum_{i=L_{sw}+2-n}^{N} P_{sw\_n}(t)*(g_{sw\_i})*(r_{sw}) \cdot \quad (14)$$

The four performance measures are as follows.

$$SW\_PPS\_Drop \quad Rate(\rho, S, t, N, K, L_{sw}) = P_{sw\_drop}(t) \cdot \quad (15)$$

$$SW\_PPS\_Throughput(\rho, S, t, N, K, L_{sw}) = \rho - P_{sw\_drop}(t) \cdot \tag{16}$$

$$SW\_PPS\_Internal\quad Delay(\rho, S, t, N, K, L_{sw}) = \frac{[\sum_{i=1}^{L_{sw}} (i) * (P_{sw\_i}(t))] + L_{sw} * P_{sw\_drop}(t)}{1 - P_{sw\_0}(t)} \cdot \tag{17}$$

$$SW\_PPS\_External\quad Delay(\rho, S, t, N, K, L_{sw}) = \frac{[\sum_{i=1}^{L_{sw}} (i) * (P_{sw\_i}(t))] * K + L_{sw} * P_{sw\_drop}(t) * K}{S - P_{sw\_0}(t) * S} \tag{18}$$

6 Speedup Requirements for FCFS-OQ Switch Emulation

If the FCFS-OQ Switch Drop Rate (Eq. 6), divided by the SW-PPS Drop Rate (Eq. 15) exceeds one, the drop rate of SW-PPS emulates that of FCFS-OQ switch. Based on the assumption that length of a FCFS-OQ switch queue equals length of center-stage switch queue, $L = L_{sw}$. *FCFS_OQ_Switch_Drop_Rate/SW_PPS_Drop_Rate≥1*. This means that $P_{drop}(t)/P_{sw\_drop}(t) \geq 1$, so that

$$S \geq 1 \cdot \tag{19}$$

Applying $S = 1$ and $L = L_{sw}$ yields $g_i \geq g_{sw\_i}$ and, $P_j \geq P_{sw\_j}$, where $0 \leq i \leq N$ and $0 \leq j \leq L$. Since $S \geq 1$, $P_{drop} \geq P_{sw\_drop}$. Throughput is determined by subtracting drop rate from arrival rate. With a speedup of $S \geq 1$ and internal link rate $\geq R/K$, the SW-PPS can emulate FCFS-OQ switch throughput and drop rate.

A same procedure demonstrates that SW-PPS can match the cell delay of FCFS-OQ switch, if the following inequality is satisfied. If F*CFS_OQ_Switch_Delay /SW_PPS_External_Delay≥1*, we have *FCFS_OQ_Switch_Delay/SW_PPS_Internal_ Delay*(K/S)≥1*. By increasing speedup S, the SW-PPS Internal Delay (Eq. 17) can be effectively reduced. We also obtain

$$S \geq \frac{SW\_PPS\_Internal\_Delay * K}{FCFS\_OQ\_Switch\_Delay} \cong \frac{K * (1 - P_0(t))}{[\sum_{i=1}^{L} (i) * (P_i(t))] + L * (P_{drop}(t))} \cdot \tag{20}$$

7 Comparison and Simulation

The evaluations used to measure performance are mean throughput, cell loss ratio, and mean cell delay. A 32*32 FCFS-OQ switch and SW-PPS are established; the number of center-stage FCFS-OQ switches is K= 64; the queue size (L) is 16 and 4. Figures 5 to 10 show a mathematical analysis and simulation results for the OQ (L, x) and SW-PPS (L, x, speedup). If x= S represents the simulation result, then x= M yields the results of the Markov chains analysis (whose results are plotted as hollow points in figures 5 to 10). When the speedup satisfies Eq. 19, figures 5 to 8 compare the throughput and drop rate for OQ (L, x) and SW-PPS (L, x, speedup= Eq. 19). The SW-PPS accurately emulates the FCFS-OQ switch. The cell delay of SW-PPS (L, x,

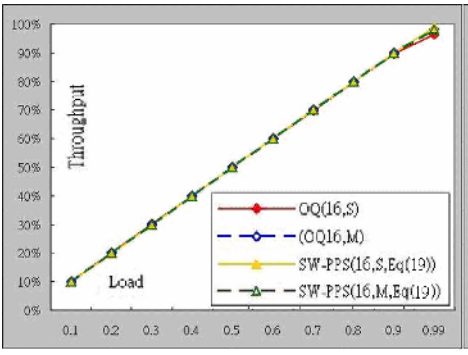

Fig. 5. Emulation of throughput (L=16)

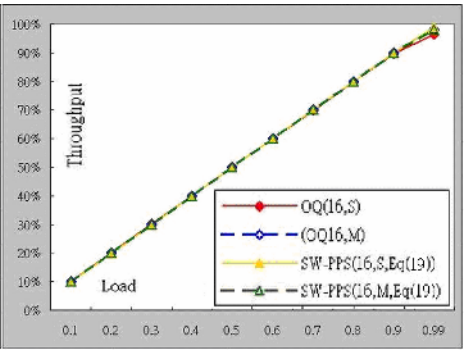

Fig. 6. Emulation of throughput (L=4)

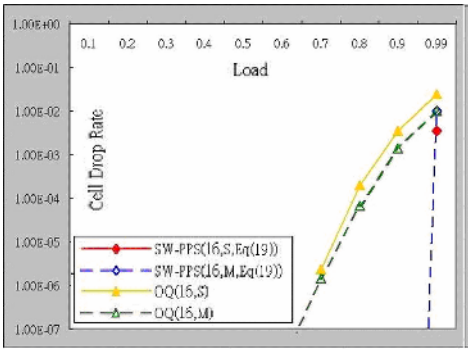

Fig. 7. Emulation of drop rate (L=16)

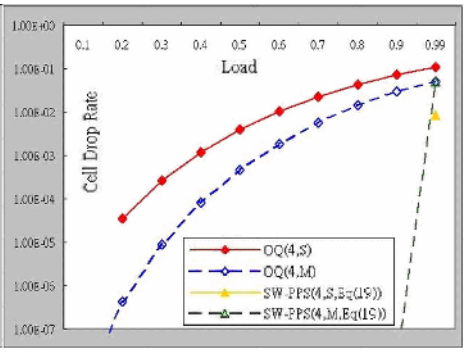

Fig. 8. Emulation of drop rate (L=4)

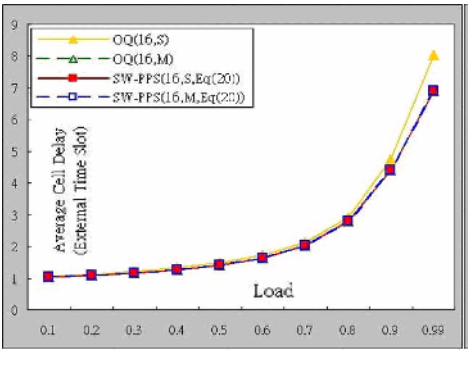

Fig. 9. Emulation of delay (L=16)

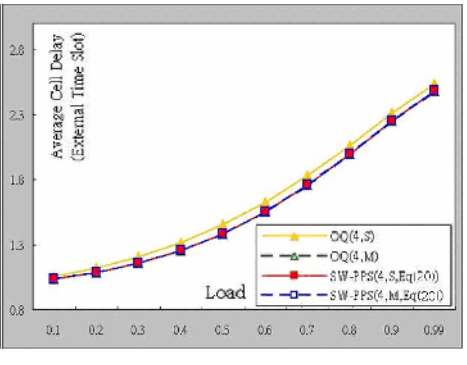

Fig. 10. Emulation of delay (L=4)

speedup= Eq. 20) is very close to that of OQ (L, x) (figures 9 and 10), when the speedup of internal link rate is as given by Eq. 20. Figures 5 to 10 reveal that the FCFS-OQ switch performance is emulated remarkably well closely. The analytical results agree closely with the simulation results.

8 Conclusion

A novel SW packet switching scheme for PPS, called SW-PPS, was presented [1]. It retains advantage of traditional PPS, that all memory buffers and internal line rate run slower than external line rate, and it uses memory space more effectively. This work develops an analytical model for measuring SW-PPS, and indicates that a SW-PPS, which operates slower than the external line rate, can emulate a FCFS-OQ switch. This analytical model exploits approximate Markov chain models. The important findings of this investigation are: (1) in throughput and cell drop rate, a SW-PPS emulates a FCFS-OQ packet switch, which is work conserving, if each slow speed packet switch works at a rate of around R/K (shown in Eq. 19); and, (2) in delay of cells, a SW-PPS can emulate a FCFS-OQ packet switch when each slow speed packet switch works at a rate of about (R/cell delay of FCFS-OQ switch) (shown in Eq. 20).

References

1. Chia Lung Liu and Woei Lin: Performance Analysis of the Sliding-Window Parallel Packet Switch. IEEE 40th International Conference on Communications 2005, ICC 2005. vol. 1 (2005). 179–183
2. Sundar Iyer: Analysis of the Parallel Packet Switch Architecture. IEEE Transactions on Networking, vol. 11, NO. 2. (2003) 314–324
3. A. Aslam, and K. Christensen: Parallel packet switching using multiplexers with virtual input queues. LCN 2002. (2002) 270–277
4. Szymanski, T., Shaikh, S.: Markov chain analysis of packet-switched banyans with arbitrary switch sizes, queue sizes, link multiplicities and speedups. Infocom 1989. vol. 3. (1989) 960–971.
5. Karol, M., Hluchyj, M., Morgan, S.: Input Versus Output Queueing on a Space-Division Packet Switch. Communications, IEEE Transactions on vol. 35. Issue 12. (1987) 1347–1356