Available at: http://publications.ictp.it IC/2007/043

United Nations Educational, Scientific and Cultural Organization
and
International Atomic Energy Agency

THE ABDUS SALAM INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS

# EXACT ANALYSIS OF PACKET REVERSED PACKET COMBINING SCHEME AND MODIFIED PACKET COMBINING SCHEME; AND A COMBINED SCHEME

Chandan T. Bhunia[1]
*SMIEEE, Haldia Institute of Technology, Haldia, West Bengal, India*
*and*
*The Abdus Salam International Centre for Theoretical Physics, Trieste, Italy.*

**Abstract**

Packet combining scheme is a well defined simple error correction scheme for the detection and correction of errors at the receiver. Although it permits a higher throughput when compared to other basic ARQ protocols, packet combining (PC) scheme fails to correct errors when errors occur in the same bit locations of copies. In a previous work, a scheme known as Packet Reversed Packet Combining (PRPC) Scheme that will correct errors which occur at the same bit location of erroneous copies, was studied however PRPC does not handle a situation where a packet has more than 1 error bit. The Modified Packet Combining (MPC) Scheme that can correct double or higher bit errors was studied elsewhere. Both PRPC and MPC schemes are believed to offer higher throughput in previous studies, however neither adequate investigation nor exact analysis was done to substantiate this claim of higher throughput. In this work, an exact analysis of both PRPC and MPC is carried out and the results reported. A combined protocol (PRPC and MPC) is proposed and the analysis shows that it is capable of offering even higher throughput and better error correction capability at high bit error rate (BER) and larger packet size.

MIRAMARE – TRIESTE

July 2007

[1] Senior Associate of ICTP. ctbhunia@vsnl.com

## I. INTRODUCTION

The Packet combining (PC) scheme for correction of bit errors using the erroneous copies of a packet at the receiver was introduced by Chakraborty[1]. In the scheme, two erroneous copies are XORed for locating the position of bit(s) in errors of a packet, so that the receiver can correct the error rather than requesting the transmitter to retransmit the packet. The correction process proposed by Chakraborty is the brute force bit by bit inversion of the located bit error positions and followed by FCS (frame check sequence) check. This process fails under the following situations:

(i) when the bit error locations in erroneous copies are same and

(ii) when the multiple bit errors occur, as then the application of brute force bit inversion for correcting will be huge and complex. For n bits in error (n>1), on average $2^{n-1}$ trails of attack are required.

A scheme called Packet Reversed Packet Combining (PRPC) has been shown to tackle the first problem of PC [2]. While the Modified Packet Combining (MPC)[3,4] was reported to tackle the multiple bit errors in the received erroneous copies of packet. MPC does not handle errors that occur at the same bit location until an odd number of erroneous copies are available and only when a transmitted bit 0 is converted to 1 in all copies in the same location but not when a transmitted bit 1 is converted to 0 in all copies in same location.

Both PRPC and MPC scheme individually or in combination are believed to offer higher throughput than that of basic ARQ protocol. Existing error correction scheme for networks mainly address the correction for single or double bit error(s) and when bit error rate is $10^{-3}$ or less, the probability of more than a double bit error in the packet is insignificant. In view of this PRPC and MPC are prominent schemes subject to offering tolerable throughput. In previous studies neither adequate investigation nor exact analysis was done to substantiate the claim of acceptable or higher throughput. In this work an exact analysis of both PRPC and MPC is carried out and a combined scheme (of PRPC and MPC) is proposed. The results obtained show that the combined scheme offers higher throughput and better error correction capability.

## II. REVIEW OF PRPC AND MPC

*PRPC*

The idea behind PRPC [2] is that when the receiver receives an erroneous packet and requests for retransmission of another copy without discarding the first erroneous copy, the transmitter transmits a bit reversed packet of the original. This is better illustrated with the examples below.

Example 1: Say the original packet is, 00110101. Say on a first transmission the receiver receives the packet as 0011**1**101 (call it first copy) (error location is marked bold, 5th bit from left). Receiver requests for retransmission. Transmitter retransmits a copy with bit reversed as: 10101100 (bit wise reversed copy of original packet, LSB of original packet is now MSB of bit reversed packet and vice versa). Say the receiver gets the bit reversed copy erroneously with error at same error location ($5^{th}$ bit from left). Thus the receiver will receive the copy as: 1010**0**100 (call it second copy).

Example 2: Say the original packet is, 01. Say on a first transmission the receiver receives the packet as 0**0** (call it first copy) (error location is marked bold). Receiver requests for retransmission. Transmitter retransmits a bit reversed copy as: 10 (bit wise reversed copy). Say the receiver gets the bit reversed copy erroneously with the same error location. Thus the receiver will receive the copy as: 1**1** (call it second copy).

Example 3: Say the original packet is, 111111111. Say on a first transmission the receiver receives the packet as 1111**0**1111 (call it first copy) (error location is marked bold). Receiver requests for retransmission. Transmitter retransmits a bit reversed copy as: 111111111 (bit wise reversed copy). Say the receiver gets the bit reversed copy erroneously with the same error location. Thus the receiver will receive the copy as: 1111**0**1111 (call it second copy).

[NOTE: to mark the bit reversed copy, we have underline the copy. This is for illustration purpose]

In the examples above, the receiver will now perform correction operation as below:

I. The receiver reverses the second copy bit wise. In the example(1), we get second copy on reversing as 0011**0**101. Now receiver does the correction as in PC with reversed second copy and first copy. In the example, XOR of first and reversed second copy will result in 00011000. Thus, now, application of brute force bit inversion on $4^{th}$ & $5^{th}$ bit will correct the error and it will require on average 2 trails only. In the example (2), XOR operation will result 11. Brute force bit inversion scheme be employed to correct.

II. In example (3) correction is not possible. This is because error is exactly at the middle bit of packet. (in example 1 & 2, each of the packets is of 8 bits. In example 3, the packet is of 9 bits. Middle bit is $5^{th}$ bit from both ends.) Bit reversion does not change its position. Thus the correction technique only works for packets with an even number of bits.

Luckily on data networks, packets are (in size) always multiple of several bytes, and thus always of an even number of bits which authenticates the no-failure case of correcting bit errors by PRPC scheme.

Unlike PC, the PRPC scheme is capable of correcting all single bit errors by using two consecutive erroneous packets even when the error occur at the same location, because the packet reversing changes the bit position. This could be stated as in PRPC, the $i^{th}$ bit from the right of original packet of k bits to $(k-i+1)^{th}$ bit in reversed packet for i=1 to k.

*MPC*

In the MPC [3.4] technique, on getting a request for retransmission from the receiver the transmitter sends i (i>1) copies of the requested packet. The receiver on getting i copies uses a pair-wise XORed algorithm to locate error positions. For example if i=2, we have three copies of the packet (Copy-1=the stored (original) copy in receiver's buffer, Copy-2=one of the retransmitted copies, Copy-3=another retransmitted copy) and three pairs for xor operation are:

Copy-1 and Copy-2
Copy-2 and Copy-3
Copy-3 and Copy-1

Table I: Algorithm of MPC

| Comparing pairs | Number of bits in error (x) | Common copy in two consecutive (x) |
|---|---|---|
| Copy-1 and Copy-2 | 1 | Copy-1 common in first two xs |
| Copy-1 and Copy-3 | 2 | Copy-3 common in next two xs |
| Copy-3 and Copy-2 | 3 | Copy-2 common in next two xs |

Assume that a transmitted packet 10100011 received erroneously at receiver on first transmission is:

Copy-1 = 10101011

and subsequently retransmitted two copies received at receiver are

Copy-2 = 10101111, and
Copy-3 = 10100001.

Under XORed operation of combination as below we have:

Copy-1 XORed Copy-2 (say, $C_{12}$) = 00000100 (one bit in error)
Copy-2 XORed Copy-3 ($C_{23}$) = 00001110 (three bits in error)
Copy-3 XORed Copy-1 ($C_{31}$) = 00001010 (two bits in error).

Now the algorithm for selecting which copy the bit inversion will start and how to proceed is based on the table (see Table (I)) in ascending order of number of bits in error as indicated by the xor operation. The bit inversion and the FCS checking process shall begin with the common copy indicated in the last column of the table so prepared, and proceed down the table if required. In this example (using table (I)), the detection of error location and consequent bit inversion will start with Copy-1 and if required will be followed by Copy-3 and then by Copy-2.

## III. THROUGHPUT ANALYSIS

*PRPC*

Basic throughput of all ARQ techniques depends on the average number (x) of times a packet needs to be transmitted (including retransmission(s)) for successful reception by the receiver. As x decreases, throughput increases, and when x=1 throughput is 100%.

this work studies the gain (in throughput) of the PRPC and MPC schemes over the normal stop and wait ARQ technique, given that the achieved gain will be equally true for other ARQ schemes namely GBN and SRQ [5-8].

Assuming the gain will be measured by the parameter called throughput, n which is defined (in literature) for the normal stop and wait ARQ as:

$$n_{sw} = \frac{1}{1-P} \quad (1)$$

where the packet error probability P $= 1-(1-\alpha)^k$ and $\alpha =$ bit error rate, and k = packet size in bits.

In the PRPC scheme, we already know that all single bit errors will be corrected successfully. The probability that a packet is with single bit error is

$$P_1 = C_1^k \alpha^1 (1-\alpha)^{k-1} \quad (2)$$

Thus the probability of a packet having more than a single bit in error is:

$$P - P_1 \quad (3)$$

In PRPC, for the correction of single bit error, a packet is transmitted twice: first the original and next the reversed. Thus PRPC when implemented in stop and wait ARQ protocol, the average number of times, $n_{prpc}$ a packet needs transmission (including retransmission) for successful delivery is:

$$n_{prpc} = \frac{P-P_1}{1-(P-P_1)} + 2.P_1 \quad (4)$$

the first part of the right hand side of eq.(4) is for the correction in normal stop and wait ARQ for the bit errors other than the single bit errors, and the second part is for PRPC in correcting single bit errors.

Simple Proof:

i) when $P=P_1=0$ (no error case, result must be 1 packet per success) we got that as $n_{prpc} = 1$;
ii) when $P=P_1=1$(all single bit error, result must be 2 packets per success as in PRPC), we got that as $n_{prpc} = 2$ and
iii) when $P=1$ and $P_1 \neq 1$ (always packet is in error but error is not always single bit error, result must be towards infinite packets per success), we got that as $n_{prpc} \to \infty$.

From eqs.(1&4), the gain of PRPC over normal stop and wait ARQ is obtained as:

$$gain = n_{sw} - n_{prpc}$$
$$\frac{P_1 - PP_1 - 2P_1(1-P)(1-P+P_1)}{(1-P)(1-P+P_1)} \quad (5)$$

Thus an analysis on gain as in eq. (5) may be made. First, the gain is obtained only when:

$$P_1 \geq [PP_1 - 2P_1(1-P)(1-P+P_1)] \quad (6)$$

Thus the gain is possible over selected sets of (k, α) i.e. (packet size, bit error rate), although in next section we will see for practical ranges of k and α, gain is always obtained, thereby duly meeting the inequality (6).

Second, eq. (5) gives absolute gain when inequality (6) is satisfied.

A more accurate parameter for gain is percentage gain and it is:

$$gain = [gain / n_{sw}] \times 100 \qquad (7)$$

%Gain over different ranges of k and α as obtained numerically over eq. (7) are shown in fig 1: We find:

i) %gain increases with bit error rate. This is due to increasingly higher single bit error rate compared to rates of double or higher bit error rate with increasing packet size,

ii) PRPC therefore offers higher throughput than that of conventional ARQ techniques for practical ranges of k and α.

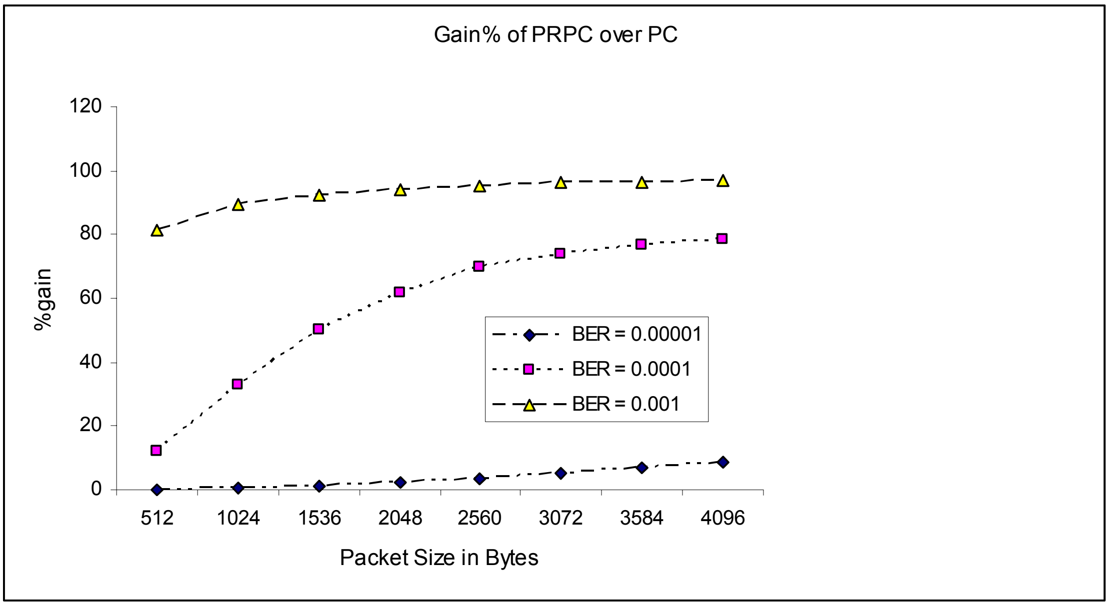


Fig 1: Variation of gain and gain% with packet size for different bit error rate

*MPC with PRPC*

We propose a MPC operation as shown in fig. 2. This is a combined scheme of MPC and PRPC, called **MPC with PRPC**. The scheme is aimed at the correction of both single bit error (with PRPC) and double bit error with (MPC) at the receiver by erroneous copies. In this scheme, when a negative acknowledgment is received, the transmitter will transmit two copies; one in PRPC mode and another copy in the original form of the original packet. Then in MPC with PRPC, up to double bit errors will be corrected at the receiver. The probability that a packet in error except with single bit error and double bit error is:

$$P - P_1 - P_2 \qquad (8)$$

where $P_2$ is the probability of a packet with double bit error, and

$$P_2 = C_2^k \alpha^2 (1-\alpha)^{k-2} \qquad (9)$$

For correction up to double bit error, a packet is transmitted 3 times: first original and then two copies (on retransmissions request). Thus in MPC with PRPC when implemented in stop and wait ARQ protocol, the average number of times, $n_{mpc}$ a packet needs transmission (including retransmission) for successful delivery is:

$$n_{mpc} = \frac{1 - P_1 - P_2}{1 - (P - P_1 - P_2)} + 3.(P_1 + P_2) \qquad (10)$$

First part of right hand side of eq. (10) is for correction in normal stop and wait ARQ for bit errors other than single bit and double bit error, and second part is for MPC with PRPC correcting up to double bit error. Thus the gain of MPC with PRPC if any over PRPC is

$$gain_{mpcprpc} = n_{prpc} - n_{mpc} \qquad (11)$$

The percentage gain is then:

$$gain_{mpcprpc} = [n_{prpc} - n_{mpc}] \times 100 / n_{prpc} \qquad (12)$$

The numerical results obtained on eqs. (11 and 12) are shown in table II. It is found that:

i) gain is achieved only when BER is high and BER ≥0.001. This is because at high bit rate, presence of double bit error is significant

ii) gain increases with packet size (fig 3), as there is more probability of having packets with double bit errors when the packet size is large.

iii) for BER less than 0.001, gain is obtained only when packet size is ≥ 2560 bytes.

Table II: %gain of MPC with PRPC over PRPC

| Bit Error rate (α) | k in bytes | %gain |
|---|---|---|
| 0.0001 | 512 | -25.65 |
| | 1024 | -34.56 |
| | 1536 | -30.62 |
| | 2048 | -16.02 |
| | 2560 | 3.947 |
| | 3072 | 23.42 |
| | 3584 | 39.01 |
| | 4096 | 50.15 |
| 0.001 | 512 | 62.58 |
| | 1024 | 78.70 |
| | 1536 | 85.05 |
| | 2048 | 88.54 |
| | 2560 | 90.71 |
| | 3072 | 92.19 |
| | 3584 | 93.27 |
| | 4096 | 94.088 |

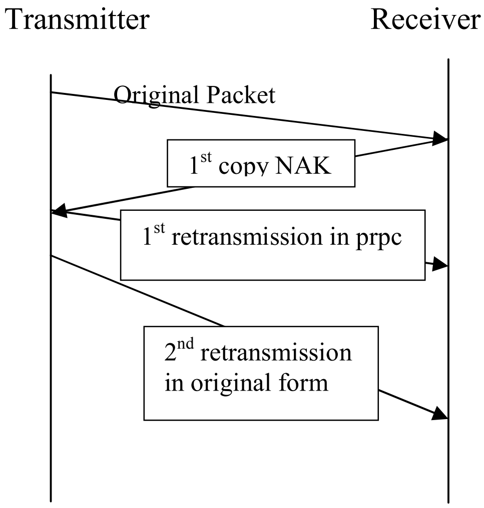


Fig. 2: MPC with PRPC in stop and wait ARQ operation

## IV PACKET ERROR CORRECTION CAPABILITY

Besides throughput, we investigate on packet error correction capability to estimate the full potentiality of proposed schemes. Towards that investigation we study the improvement in terms of gain in packet error correction probability achieved in the schemes.

*PRPC*

If a packet is made of k bits, then the probability that the consecutive two packets get error in same location is the probability that two consecutive error vectors are same. This probability equals to $\frac{1}{k^2}$. With PC, the correction of single bit error if occurs at same location of erroneous packets is not possible. PC therefore has the probability of single bit error correction as:

$$p_{cpc} = P_1\left(1 - \frac{1}{k^2}\right) \tag{13}$$

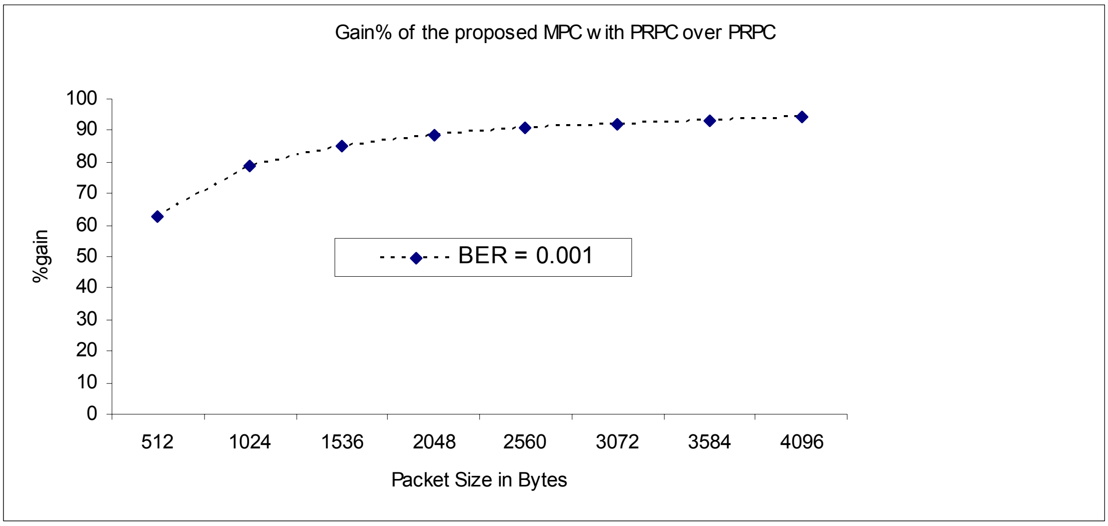


Fig 3: %gain of MPC with PRPC over PRPC

With PRPC, all single bit error corrections are made with erroneous copies having its correction probability of packet error equal to $P_1$. Thus percentage gain in packet error correction probability of PRPC over PC:

$$gain = [P_1 - P_1(1-\frac{1}{k^2})] \times 100 / [P_1(1-\frac{1}{k^2})]$$
$$100/(k^2 - 1) \tag{14}$$

Eq. (14) shows that gain% is independent of bit error rate which should be the case as the probability of occurrence of the same error vector in erroneous copies is independent of bit error rate. Fig 4 is drawn using numerical results on eq. (14). It shows that

i) gain% is always present for all packet sizes and
ii) it decreases with packet size.

This is because the probability of two erroneous packets having same error vectors also decreases with increasing packet size.

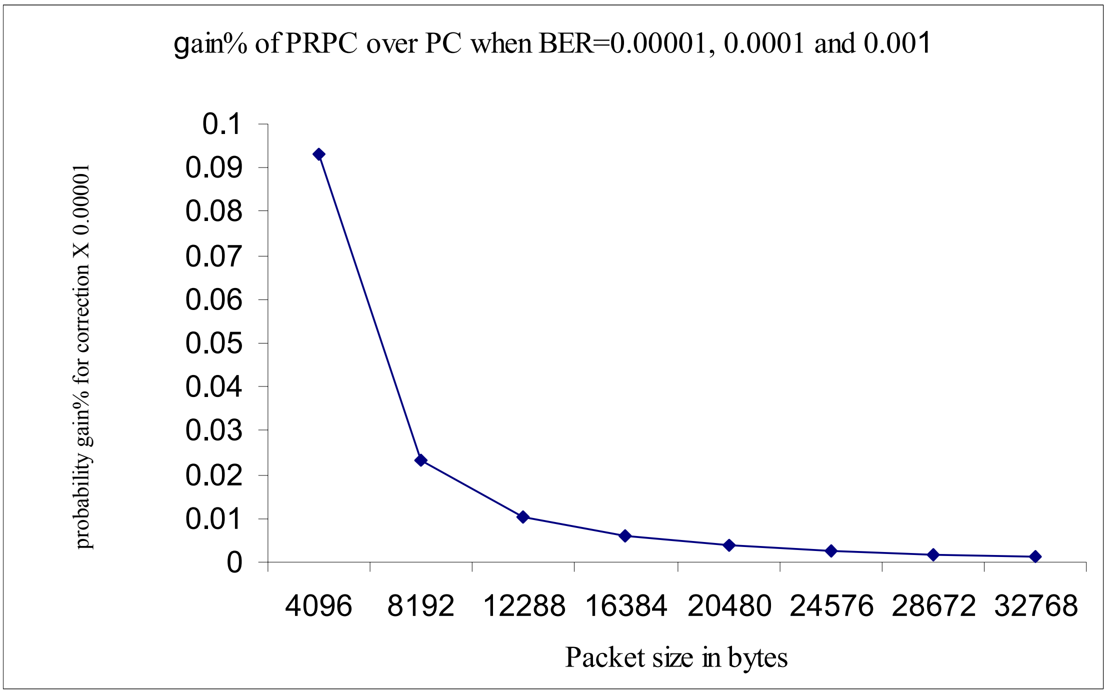


Fig 4: gain% in probability of error correction in PRPC over PC

*MPC with PRPC*

PRPC corrects all single bit error in packet. MPC with PRPC corrects up to double bit error in packet. Then the probability gain in correcting packet by MPC with PRPC over PRPC:

$$gain = \frac{P_2}{P_1} \times 100 \tag{15}$$

The numerical results of eq. (15) is shown in fig 5. We find:

i) gain% is there over all packet sizes and bit error rate, and
ii) gain% increases with packet size as well as bit error rate, thereby justifying the superiority of the technique over other techniques.

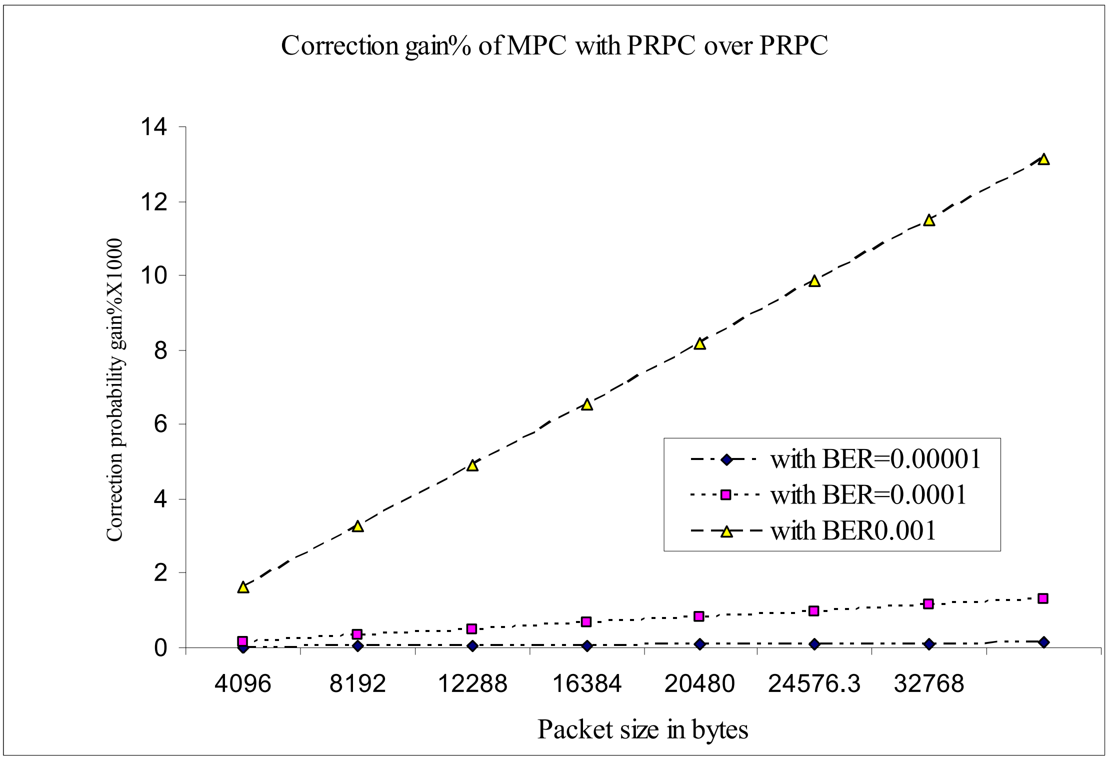


Fig 5: gain% in probability of error correction in MPC with PRPC over PRPC

## V. CONCLUSION AND FUTURE RESEARCH

A comparative study of PC, PRPC and MPC with PRPC with normal stop and wait ARQ protocol is made in terms of throughput and packet error correction probability. It is found that the proposed PRPC and MPC with PRPC are respectively better solutions for links with only single bit error and links with up to double bit error. Both the proposed schemes offer better throughput compared to normal ARQ techniques. In future, the proposed MPC with PRPC will be studied in comparison to other modified ARQ schemes [6, 9-12].

**Acknowledgments.** This work was done within the framework of the Associateship Scheme of the Abdus Salam International Centre for Theoretical Physics, Trieste, Italy.

## REFERENCES

[1] Shayam S Chakraborty,, “An ARQ scheme with packet combining”, IEEE Communication Letters, Vol 2, No 7, pp200-202, July’1995

[2] Chandan T Bhunia, Packet Reversed Packet Combining Scheme, Pre print ICTP/2006/46 046, pp1-7, Italy

[3] Chandan T Bhunia, “Modified packet Combining Scheme” Proc. ICITA, IEEE Comp. Soc., Vol 2, pp 641-646, 2005

[4] Chandan T Bhunia, “Information Technology, Networks and Internet”, New Age International Publishers, New Delhi, 2005

[5] Shyam S Chakraborty, “An Adaptive ARQ scheme with Packet Combining for time varying channels”, IEEE Communication Letters, Vol 3, No 2, pp 52-54, Feb’1999

[6] Chandan T Bhunia, “ARQ-Review and Modifications”, J IETE Tech Rev. Vol 18, pp381-401, Sept’2001

[7] Thomas Kostas and Scott Jordan, “Packet Erasure FEC on ARQ protocols”, University of California, 544 D

[8] William G Chambers, Basics of Communication and Coding, Oxford Science Publications, 1985, Ch 6

[9] Chandan T Bhunia. New and Modified ARQ Schemes and their performance analysis, Int’l J HIT Transaction on ECCN, No 1, Vol 1, pp1-26, 2006, HIT , India

[10] E J Weldon Jr, An Improved Selective Repeat ARQ Strategy, IEEE Trans Comm, Vol 30, No 3, March'82, pp 480-486

[11] Yu Dong Yao, An Effective Go-Back-N ARQ Scheme for Variable Error Rate Channels, Vol 43, No 1, Jan’95, pp 20-23

[12] Michele Zorzi and Ramesh R Rao, Lateness Probability of a Retransmission Scheme for Error Control on a Two-State Markov Channel, IEEE Transactions on Communications, Vol. 47, No. 10, October’1999, pp.1537-1548