INDIAN INSTITUTE OF MANAGEMENT CALCUTTA

WORKING PAPER SERIES

WPS No. 662/ September 2010

Efficient Management of Lightpaths in WDM Optical Networks Employing Multiple Wavelengths Concurrently during Setup

by

Malabika Sengupta
Dept of Computer Science and Engineering, Kalyani Government Engineering College, Kalyani, West Bengal, India

Swapan Kumar Mondal
Dept of IT, Kalyani Government Engineering College, Kalyani, West Bengal, India

&

Debashis Saha
Professor, IIM Calcutta, Joka, Diamond Harbour Road, Kolkata 700104, India

Efficient Management of Lightpaths in WDM Optical Networks Employing Multiple Wavelengths Concurrently during Setup

Malabika Sengupta[1], Swapan Kumar Mondal[1] and Debashis Saha [2]

[1]Kalyani Government Engineering College, Kalyani, Nadia, INDIA,

[2]Indian Institute of Management, Calcutta, INDIA

Abstract: During lightpath establishment in WDM optical networks, two important steps, other than routing, are: wavelength selection and wavelength reservation. During wavelength reservation, often multiple connection requests unknowingly compete for the same wavelength, even when other free wavelengths are available, resulting in a collision. Attempt of multiple wavelengths reservation may improve the probability of successful reservation. This aggressive reservation is used on Destination Initiated Reservation Protocol (DIRP) and the scheme is reported as Destination Initiated Multi-wavelength Reservation Protocol (DIMRP). To reduce the vulnerable period, concept of dynamic splitting is used to initiate reservation from intermediate node. The dynamic splitting is implemented in Split Reservation Protocol (SRP). Aggressive reservation is used on SRP and performance is improved considerably. SRP is further improved by intelligently guessing a wavelength in advance, using Markov model, so that wavelength conflict may be effectively reduced. Even then a connection request may be blocked because of the vulnerable period between wavelength guessing and actual reservation. To minimize the effect of such vulnerability, *multi-wavelength* approach is used, and the protocol is reported as Multi-wavelength Markov based Split Reservation Protocol (MMSRP). MMSRP basically handles *multi-wavelengths* through multiple splitting. Simulation results show that the *blocking probability* (*bp*) in MMSRP decreases considerably compared to MSRP. In this paper, different schemes using *multi-wavelength* approach are discussed separately and performance of individual protocols are compared with contemporary related protocols. Also we have done the comparative study among those protocols. From the study it is found that, in general, *multi-wavelength* approach improves the performance and among the mentioned protocols, MMSRP performs best in respect of *bp*.

Index Terms— **Optical networks, WDM, wavelength reservation, protocols, splitting, Markov model, blocking probability, delay.**

I. INTRODUCTION

In distributed WDM optical networks [1]-[4],[11],[15]-[17] having no wavelength conversion facility [5], usually a dedicated lightpath is first established between the *source-destination* pair, before the actual data transfer starts. A continuous path, having the same wavelength reserved in all the hops of the path, is called a lightpath. Lightpath establishment [1] involves three basic steps: (i) routing, (ii) wavelength selection and (iii) wavelength reservation. Here we have considered fixed shortest path routing. However the protocols discussed here may use other routing methods also. The method of selection of wavelength (for reserving it later) is very important because it indirectly affects the sharing of wavelengths (a critical resource in WDM networks) and hence blocking of requests. Requests are normally blocked due to non availability of wavelengths. But blocking may also occur due to collision, when two or more requests try to reserve the same wavelength without noticing the other free wavelengths.

Though different methods are used for selection of wavelength for reservation, two conventional methods are: *random-fit* and *first-fit* [5]. In *random-fit*, a wavelength is selected randomly from the available pool of wavelengths. In *first-fit*, all wavelengths are indexed in an order, and the wavelength having the lowest index is selected from the available set of wavelengths for reservation. In another method [10], wavelength selection is done using label prioritization [12], where the priorities of wavelengths are set depending on their duration of stay in the pool. Another important selection method is using Markov model. A notable scheme, Markov based Backward Reservation Protocol (MBRP) [7], has used this method for selection of wavelength.

In MBRP, selection of wavelength is done using Markov method and reservation is done using Destination Initiated Reservation Protocol (DIRP). Here, a wavelength is guessed for a particular request well in advance, so that other requests do not select that guessed wavelength. Thus, wavelength conflict among contemporary requests is reduced. Consequently, MBRP performs better than DIRP [10]. In Markov based Split Reservation Protocol (MSRP), wavelength selection is done using Markov model and reservation of wavelength is done using Split Reservation Protocol (SRP). Using the concept of splitting [7], blocking is reduced by shortening the *vulnerable period*. So MSRP performs better than MBRP [8]. However, a major limitation of MSRP is that, it attempts only one wavelength for reservation and splitting is done only once. If the selected wavelength fails during reservation, the request is blocked. Hence, there is still space for further improvement by extending Markov selection to *multiple wavelengths* and subsequently incorporating *multiple splitting*. This concept helps many erstwhile failure cases succeed, thereby reducing the overall blocking of requests. This protocol is reported as Multi-wavelength Markov based Split Reservation Protocol (MMSRP) [14].

Successful reservation of selected wavelength is also very crucial. We know that, information about wavelength availability is difficult to be guaranteed at any particular place and time in a large distributed system like WDM network. Initially, to handle this issue, two basic reservation protocols, were suggested: Source Initiated Reservation Protocol (SIRP), which is also called Forward Reservation Protocol (FRP), and DIRP, also known as Backward Reservation Protocol (BRP).

For lightpath establishment, in forward reservation protocols, reservation is initiated from *source*. So reservation is done much before the wavelength is actually used for data transmission. That increases the *reservation duration*. The duration for which wavelength is reserved prior to actual data transfer, is called, *reservation duration*. If *reservation duration* is increased then wavelength is reserved for a longer period of time which increases *over reservation*. Due to *over reservation* other requests may not get the free wavelengths, and, hence, overall blocking increases. To reduce *over reservation*, backward reservation protocol was proposed [2] where reservation is initiated from *destination* after successful probing from *source* to *destination* (*forward path*). But successful probing does not always guarantee the availability of wavelength during reservation from *destination* to *source* (*backward path*). Probing is done to check the availability of wavelength(s). Normally, a particular wavelength, which was available during probing, may not be available, while being attempted for reservation. The selected wavelength may be occupied by some other request within this interval between probing and attempt of reservation. The interval between probing and attempt of reservation is known as *vulnerable period*. The uncertainty in availability of wavelength during reservation increases with increase in *vulnerable period*. To reduce the *vulnerable period*, wavelength(s) may be reserved much earlier in the *forward path*, but this again leads to *over reservation*. Thus, reservation should be done in such a way that both *vulnerable period* and *reservation duration* are optimised. This is the most important challenge for a reservation scheme. In SIRP reservation is initiated from *source*, causing high *reservation duration*, whereas *vulnerable period* is nil. In DIRP probing is done in the *forward path*, and reservation is initiated from *destination* causing less *reservation duration* but more *vulnerable period*. Thus *vulnerable period* plays an important role in the protocols. It is reported that DIRP performs better than SIRP.

The basic limitation of DIRP is its long *vulnerable period*. On the other hand, the main disadvantage of SIRP is *over reservation* due to longer *reservation duration*. To minimize the effect of both, reservation of wavelength may be initiated from *intermediate node* and such protocols are referred in general as, Intermediate Node Initiated Reservation Protocol (INIRP). If initiation of reservation is done statically from some special nodes, the scheme is reported as IIRP (Intermediate node Initiated Reservation Protocol)[6]. It is also reported that IIRP performs better than DIRP. But in case of static INIRP, for all requests, reservation is attempted from *intermediate node*, and hence may invite *over reservation*. Thereafter dynamic INIRP is implemented and the scheme is reported as SRP. In this scheme, reservation is initiated from *intermediate node*, dynamically only when required. It is also reported that SRP performs better than IIRP, as far as *blocking probability* (*bp*) is concerned [8]. The scheme SRP is further improved by using Markov model for selection of wavelength and the scheme is reported as MSRP. It is also reported that MSRP performs better than its peers.

In Markov method, before final selection of wavelength, a wavelength is guessed for a particular connection request well in advance, so that other requests do not select that guessed wavelength. Thus, wavelength conflict among contemporary requests is reduced. In MSRP, wavelength selection is done using Markov model and reservation of wavelength is done using dynamic splitting, like SRP [8]. The concept of splitting shortens the *vulnerable period* and hence reduces the probability of blocking.

Another important aspect for wavelength reservation is the number of wavelengths attempted for reservation. This approach is referred as *multi-wavelength* reservation. Multi-wavelength reservation scheme is used to increase probability of reservation of at least one wavelength successfully, throughout the route. But this scheme reserves more resource (wavelength) and causes scarcity and hence blocking of other concurrent requests. Thus, the number of wavelengths to be reserved (*aggressiveness*) is also needed to be optimized.

Multi-wavelength approach of reservation of wavelength is used on DIRP, and the scheme is reported as Destination Initiated Multi-wavelength Reservation Protocol (DIMRP). It is already mentioned that, long *vulnerable period* causes more blocking of requests. In DIMRP, multiple free wavelengths (*b*), subject to availability, are attempted for reservation to implement *aggressiveness*. By doing so, chances of getting at least one wavelength throughout the path, is improved considerably and hence *bp* is reduced. However, this concept invites *over reservation* because, too much network resource may be used (through reservation of multiple wavelengths) by one request. In such cases, future requests may be blocked due to non-availability of wavelengths.

Aggressive reservation is also used on SRP, which is basically, dynamic INIRP. In this scheme, during probing, splitting may take place depending on some network parameters. If splitting takes place, two reservation packets are initiated in both forward and backward direction, simultaneously. Among them, forward reservation packet attempts to reserve multiple wavelengths (*b*), to implement *aggressiveness*. If backward reservation packet fails at some node, it releases the reserved wavelength and retries with another wavelength from the *reserve_ set* of forward reservation packet.

Multi-wavelength guessing is used on MSRP. In this scheme, an ordered set of wavelengths obtained using probabilistic method. This set is also continuously updated during probing in the *forward path* before first splitting. If the wavelength attempted for reservation is failed, then it uses the top candidate from the ordered list for further splitting.

These three aggressive schemes are studied in this paper and their performances are also compared.

II. VARIABLES AND CONTROL PACKETS USED

We define the following terms to be used in the subsequent sections of the paper.

R: a route consisting of *n* number of nodes: $node_0$ to $node_{n-1}$ (n >1).

Source: the first node of a route *R*, where a request comes.

Destination: the last node of a route.

Intermediate node: any node except the *source* and *destination* of a route *R* .

Present node: the node (say $node_k$) of a route where the control packet under consideration has reached. It may be noted that any node of a route may become a *present node* at a given point of time.

Next node: the node next to the *present node* i.e., $node_{k+1}$ if the movement of the control packet is considered towards *destination* or $node_{k-1}$, if the movement of the control packet is considered towards *source*.

Previous node: the node previous to the *present node* i.e., $node_{k-1}$ if the movement of the control packet is considered towards *destination* or, $node_{k+1}$ if the movement of the control packet is considered towards *source.*

Present link: the link which connects the *present node* and the *next node* of a route.

Previous link: the link which connects the *previous node* and *present node* of a route.

Some tables are also used which are maintained by each node of the network in some protocols. These are *node_table, node_link_status_table* and *markov_table.*

A *node_table* keeps records of all requests passing through the node. Each record of *node_table* contains the following attributes for route *R*.

connection_id: identity number of the request.
source_id: identity number of *source* of *R.*
destination_id: identity number of *destination* of *R.*
pre_hop_id: identity number of *previous node* in *R.*
next_hop_id: identity number of *next node* in *R.*
arrival_time: time when the request arrives at the *present node.*
guessed_wavelength: the wavelength guessed in the *present link* by the request at *present node.*

The duration of a record in a *node_table* is bounded by the estimated *source-destination* round trip time of the concerned request.

We assume that each node broadcasts its adjoining link usage information at every T seconds [7]. This link usage information is stored in *node_link_status_table* at every node. The records in *node_link_status_table* contains the attributes:
link_id: identity number of a link.
bit_map: represents status of usage of all wavelengths of the link. A '1' is placed in the *bit_ map* when the corresponding wavelength is free and a '0' otherwise. The size of *bit_map* equals to the number of wavelengths used in the links.

Markov_table contains the information of rate of change of states of the wavelength usage for all the wavelengths in all the links. The records of *Markov_table* contains the following:
link_id : identity number of a link.
rate_map: contains rate of change of states of the wavelength usage for all the wavelengths in the corresponding *link_id.*

Different control packets used in the protocols discussed here are described below. All the control packets contain following common fields: *source_id, destination_id, route_path* and *connection_id*. Where *route-path* is the ordered list of nodes on the selected route. So these fields are not mentioned again while packets are described.

PROB: moves from *source* towards *destination*, It contains the additional field: *prob_set, which* is an array indicating the availability/unavailability of each wavelength in the route. MSRP and MMSRP use one more field called *prev_guess_index* which stores the guessed wavelength.

RES_FWD: moves towards *destination* to reserve the selected wavelength(s) available as *reserve_set* in it.

RES_BKD: moves towards *source* to reserve the selected wavelength(s) available as *reserve_set* in it. It also may contain a field *future_guess_set* which contains the wavelengths marked for future reservation , if required.

ACK: moves towards *source*, caries acknowledgement of RES_FWD.

NACK: moves towards *source*, caries not acknowledgement of RES_BKD or PROB.

NACK_REL: moves towards *source* to release the reserved wavelength(s), caries not acknowledgement of RES_FWD.

REL_FWD: moves towards *destination* to release the reserved wavelength(s).

REL_BKD: moves towards *source* to release the wavelength(s) reserved so far if RES_FWD fails, caries not acknowledgement.

III. MULTI-WAVELENGTH RESERVATION IN DIRP

In DIMRP, when a request arrives, *source* initiates the usual control packet PROB, which moves towards the *destination*. If PROB reaches *destination* successfully, then *destination* checks the availability of wavelength(s) in *prob_set* of PROB. If the number of available wavelengths in *prob_set* is α (say), *destination* randomly selects min (*b*,α), number of wavelengths else selects all α wavelengths. At *destination*, *prob_set* is converted into *reserve_set*. Then RES starts from *destination*, and moves towards *source*, carrying *reserve_set*. RES attempts to reserve wavelength(s) included in *reserve_set,* at every link present on its way. If on the way, at any link, a wavelength is not available, the wavelength is dropped from *reserve_set*. If RES successfully reaches the *source* with nonempty *reserve_set*, then it is a case of success. If the number of wavelengths present in *reserve_set* is *p*,

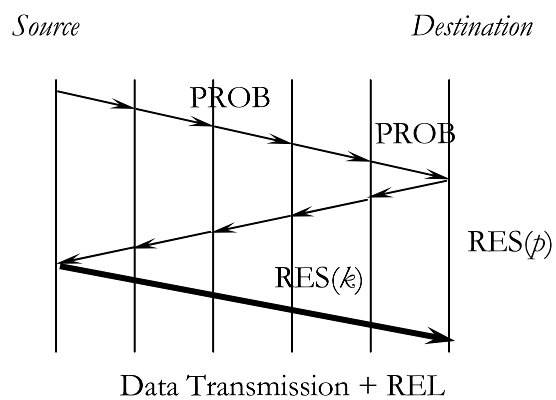

Fig.1: Case of success in DIMRP, $p<=b$ and $k<p$

(where $p<=b$) then from *p*, *source* selects one wavelength randomly for data transmission. Then RES is converted into DATA_TRANS, which transmits data and also releases the extra reserved wavelength(s) (if any), throughout the route. After completion of data transfer, the wavelength used for data transfer is released. The timing diagram presented in Fig.1, describes this case of success.

During probing, if *prob_set* becomes empty, then PROB is converted into NACK, which moves towards *source*. After the NACK reaches the *source*, the request is blocked.

Again, during reservation, if the *reserve_set* becomes empty due to non-availability of wavelength, at some *intermediate node*, then RES is converted into REL, which moves towards *destination* and releases the reserved wavelengths. Also from that node, a NACK is generated which moves towards *source*. When the NACK reaches the *source*, the request is blocked. This case of failure, during reservation is shown in Fig.2.

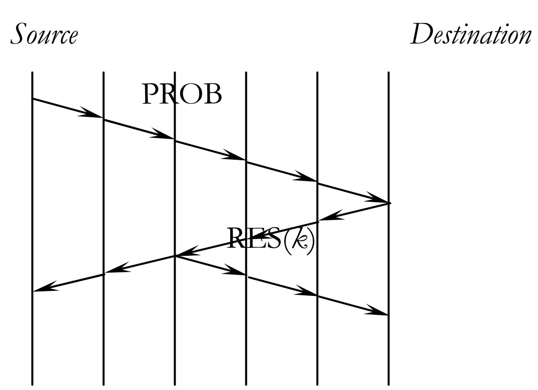

Fig.2: Case of failure during reservation in DIMRP, $p<=b$ and $k<p$

It may be noted that *multi-wavelength* (*aggressive*) reservation approach must be optimized. *Aggressiveness* (*b*) may vary from 1 to *wl* (where *wl* is total number of wavelengths in any link of the network). If *b*=1, the probability of successful reservation remains low, on other hand, if *b*=*wl, over reservation* may spoil the advantage of *multi-wavelength* reservation. Thus finding an optimum value of *b*, is a challenging issue. The optimum value of *b* depends on mean arrival rate of connection requests (*cr)* and *wl*. Henceforth we will use the word requests to represent connection requests.

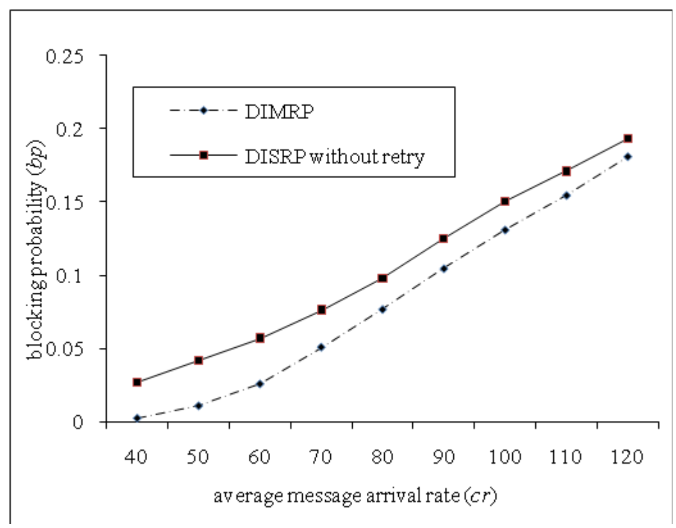

Fig.3: Comparison of *bp* with *cr* for DIMRP with other schemes for *wl*=75

Using the simulation results, DIMRP is compared with standard DIRP, also referred as Destination Initiated Single wavelength Reservation Protocol (DISRP). Fig.3 shows variation of *bp* with *cr* for *wl*=75. The figure shows that, DIMRP is a clear outperformer over DISRP. This is because success rate of reservation is improved through *multi-wavelength* reservation attempt in DIMRP.

IV. MULTI-WAVELENGTH RESERVATION IN INIRP

In general, for INIRP, *vulnerable period* is reduced at the cost of *reservation duration*. In static INIRP, reservation of wavelength is attempted unconditionally from some predefined nodes. This suffers from extreme cases because reservation from an intermediate special node is initiated unconditionally and cannot utilize properly the benefit of reduction of *vulnerable period*. For example, for a particular request, if a special node exists next to the *source* of the route, it initiates backward reservation after traversing one hop only then the case is almost like SIRP. Hence, it suffers from *over reservation* resulting increase in *bp*. Similarly, if the first special node exists just before *destination*, for a particular route, it initiates the fast RES when only one hop is left and thus probability of getting any free wavelength is reduced due to long *vulnerable period*. In dynamic INIRP i.e., SRP, the nodes, from where the initiation of reservation takes place, are not predefined, rather decided dynamically.

To reduce the *vulnerable period* and hence the *bp*, concept of splitting is introduced. Splitting means, splitting of PROB into two RES (RES_FWD and RES_BKD) at some *intermediate node*, so that reservation can be attempted in both directions (towards *source* and *destination*) simultaneously. Fig.4 shows an example of splitting.

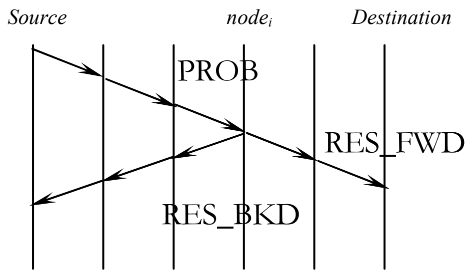

Fig.4. SRP - occurrence of splitting.

However, splitting also invites certain degree of *over-reservation,* because reservation in forward direction is done much before it is used for transmission of data. It may be noted that, if splitting occurs nearer to *destination*, *vulnerable period* increases for backward reservation (towards *source*) but the *over-reservation* in forward direction (towards *destination*) becomes less. On the other hand if splitting occurs near *source*, *vulnerable period* for backward reservation decreases but causes more *over-reservation* for forward reservation. Thus position of splitting is to be optimized in order to reduce the effect of both *over-reservation* and *vulnerable period*.

So the node where the splitting takes place may be decided dynamically using appropriate system parameters. This type of splitting is called dynamic splitting and study of the overall effect of *dynamic splitting* on its efficiency seems to be very important. The protocol is explained below.

When a request comes, *source* initiates a PROB, which moves towards the *destination*. PROB includes *hop_count* and *b* along with other fields. The variable *b* is a predefined positive integer. Basically *b* is the *aggressiveness* of the scheme, which is used for taking decision of splitting. For the first link, *prob_set* is initialised to the wavelengths available in the first link. For the subsequent links, on receiving PROB, a node performs two tasks:

- updates the *prob_set* using the operation,
  prob_set = *prob_set* $\cap$ available wavelength(s) on the *present link* and

- checks the conditions for probable splitting.

SRP dynamically splits probe attempt, into two concurrent (one towards *source* and the other towards *destination*) reservation attempts, at any *intermediate node*. For a request, splitting may occur, if the following two conditions are satisfied:

- $(x_1 * d) \le hop\_count \le (x_2 * d)$ i.e., whether the PROB has traversed more than a pre-selected distance $(x_1 * d)$ of the route as well as less than another preselected distance $(x_2 * d)$, where d is total number of hops of the route and x_1 and x_2 are two positive fractions within 0 and 1.

- number of available wavelengths of *prob_set* $\le b$, for $b \ge 1$. SRP attempts (b-1) number of retries (if required) for b>1.

If conditions of splitting are satisfied, splitting occurs. PROB is converted into two reservation packets: RES_FWD and RES_BKD. The node where PROB splits, is called the *splitting point* (*sp*). At *sp*, *reserve_set* is copied into both RES_FWD and RES_BKD. RES_BKD selects one wavelength (say w_1) randomly from the *reserve_set* and moves towards *source*, and RES_FWD moves towards *destination*, attempting to reserve all wavelengths of *reserve_set*. However, if the *reserve_set* is empty, request fails and subsequently the request is blocked.

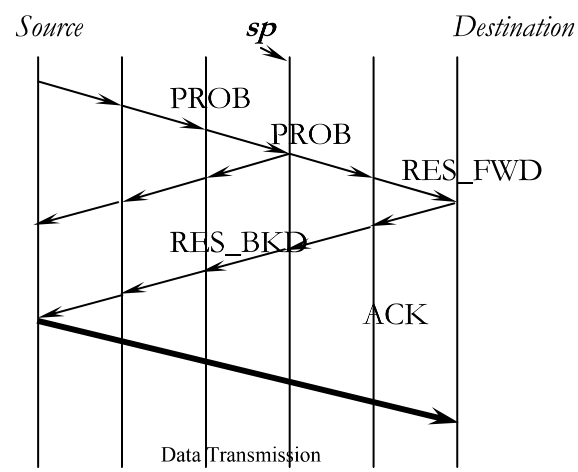

Fig.5: Case of success in SRP

If RES_BKD successfully reaches the *source*, then at *source*, it waits for ACK of RES_FWD. If RES_FWD reaches the *destination*, with nonempty *reserve_set*, then an ACK is sent towards *source* along with the *reserve_set*. ACK, on its way,

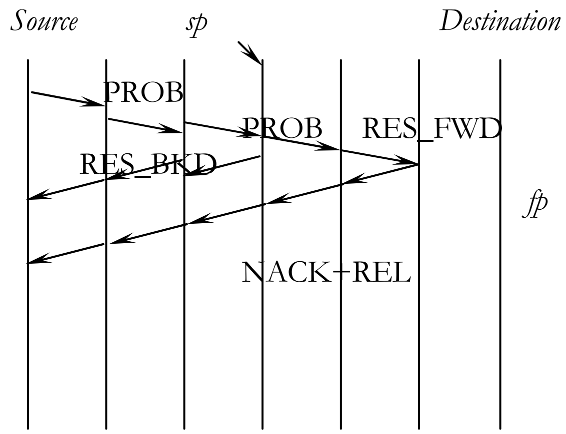

Fig.6: Failure of RES_FWD in SRP

keeps a copy of the *reserve_set* at *sp*. After receiving the ACK, the *source* checks the matching of the wavelength reserved in forward and backward directions. If those are matched, the data transmission starts (Fig.5). If there be mismatch in wavelength reservation or if RES_BKD fails, then RES_BKD is converted into REL_FWD which moves towards *sp* releasing the reserved wavelength. At *sp*, REL_FWD randomly selects another wavelength from the *reserve_set* for retry, and becomes RES_BKD again. This is repeated (if required) until total number of retries (= *b*-1) is exhausted. If RES_FWD is stuck before *destination*, then it is the case of failure (Fig.6) and it is converted into NACK_REL. The NACK_REL moves from the *intermediate node* to the *source*, and releases the wavelength reserved by both RES_FWD (from the node where failure takes place to *sp*) and RES_BKD (from *sp* to *source*). After receiving the NACK_REL at *source*, the request is blocked.

Performance of SRP is compared with its peer IIRP. Single wavelength is used for reservation in SRP, to keep parity with IIRP, and the simulation results obtained is shown in Fig. 7. From the figure, we find that, for both the schemes, *bp increases* with *cr* due to increase in crisis of wavelength. Also*,* we see that, SRP outperforms IIRP with respect to *bp*. Thus, the protocol, SRP can be considered as better performer than IIRP with respect to *bp*.

V. USE OF MULTIPLE WAVELENGTHS IN MSRP

In MSRP, when a request arrives at a node, the node guesses a wavelength based on the link usage information of the *previous link* and the *markov_table*. The wavelength thus guessed has the maximum probability of remaining available throughout the route, at that instant of time. Thus when the *source* initiates a PROB, the PROB moves towards *destination*, and each node after receiving the PROB, performs the following major tasks for the request: (i) detects the wavelengths already guessed by earlier requests and excludes them from *prob_set*, (ii) guesses a wavelength for this request from the remaining free wavelengths and updates PROB, (iii) initiates on-demand splitting (dynamically) if necessary.

MSRP adaptively splits a probe attempt into two concurrent (upstream and downstream) reservation attempts at some *intermediate node* selected dynamically. For a request, if *hop_count* is the number of hops traversed by the PROB, then, splitting may occur provided both the following conditions are satisfied:

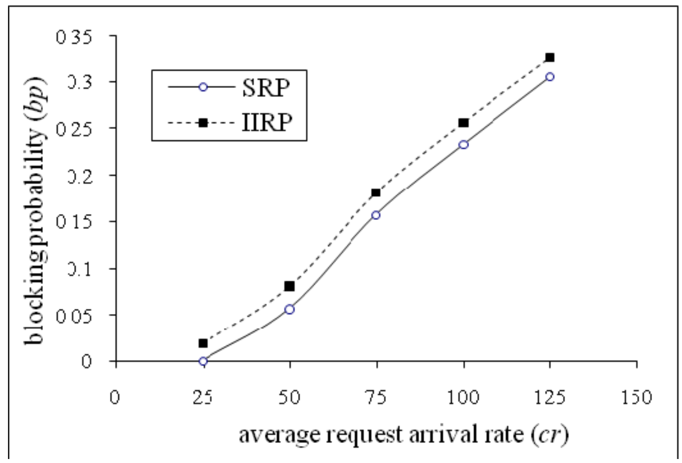

Fig.7: Variation of *bp* with *cr* for *wl*=75

(i) $(x_1 * d) \leq hop\_count \leq (x_2 * d)$ i.e., whether the PROB has traversed more than a pre-selected distance $(x_1 * d)$ of the route as well as less than another preselected distance $(x_2 * d)$, where d is total number of hops of the route, x_1 and x_2 are two positive fractions within 0 and 1, and $x_2 > x_1$.
(ii) the wavelength at *prev_guess_index* is different from λ_{gi}.

If the conditions of splitting are satisfied, splitting occurs; otherwise the PROB propagates to the *next node*.

The variation of x (i.e., x_1 and x_2) is studied in [8]. We select, x_1= 0.5, x_2= 0.6, so that *vulnerable period* as well as *reservation duration* are optimized to have low *bp.*

Two types of broadcasts are used in this protocol: (i) each node broadcasts its adjoining *link usage* information at every T seconds. This *link usage* information is stored at every node. (ii) Broadcast of *link usage* information as mentioned above, is not necessarily correct at an arbitrary time between sT and $(s+1)T$. Where s is a positive integer. To overcome this uncertainty, a prediction is suggested to select wavelength during these intervals. To take the *probabilistic* method of selection, a C-T Markov chain is used in this work. The required parameters are broadcast at every T ' seconds and stored in a table referred as *markov_table* at all nodes. So essentially *markov_table* contains the information of rate of change of states of the wavelength usage for all the wavelengths in all the links. *T '* is considered to be much longer compared to *T.* If value of *T '* is lower than a certain level, it is vulnerable to oscillation which may ultimately lead to poor performance.

Since T_{ratio} (the ratio of *T '* to *T*) is an important parameter and affects the performance of the protocol, is studied for different set of values of *cr* and *wl*. It is found that an optimum value of T_{ratio} exists in each case. It is reported that [7],[9] values of T_{ratio} corresponding to minimum value of *bp* is near 300. Hence, for simulation results, the optimum value of T_{ratio} is kept as 300.

Multi wavelength approach is used on MSRP and the scheme is referred as MMSRP. Since Markov model uses Markov chain to describe each state of wavelength usage, so maximum allowable transition is one. Thus, multiple number of wavelengths cannot be reserved at a time. In MMSRP, a set of wavelengths (instead of one) is selected by Markov model and continuously updated for possible future use. In case of failure, during reservation in the backward direction, it retries to reserve the next best wavelength, through another splitting at the failure point. Thus, MMSRP handles multiple wavelengths sequentially through *multiple splitting*. The protocol is discussed here.

In MMSRP, we extend the concept of guessing [7],[9] used in MSRP. If total number of free wavelengths is y, the node selects *b* (y>*b*>1) number of wavelengths having higher probabilities of remaining free. These *b* wavelengths are arranged with respect to probability in descending order as λg_1 ,λg_2 , …,$\lambda g b$. Here, *b* is a predefined number which represents the maximum number of splitting permitted for a request. Obviously, if y<=*b*, all y wavelengths are selected. From the ordered set, wavelength λg_1 is selected as guessed wavelength and wavelengths λg_2 to $\lambda g b$ are stored in *future_guess_set.*

When the *present node* is the *source*, *prev_guess_index* of PROB is initialized to λ_{g1}. A record is created in *node_table* of the *source* and PROB is forwarded to *next node*. If the *present node* is any node other than *source*, the node checks the availability of the wavelength stored in *prev_guess_index*. If the wavelength is available, a record is created in *node_table* of the *present node* and PROB is forwarded to *next node*; else the node checks for splitting. When splitting does not occur, *prev_guess_index* is updated to the wavelength λ_{g1}. A record is created in *node_table* of the *present node* and the PROB is forwarded to *next node*.

If splitting occurs, the PROB is converted to RES_FWD. A RES_BKD is also generated at the first *splitting point* (we call it sp_1) as shown in Figure 8 and Figure 9. RES_BKD includes the fields: *connection_id, selected_wavelength* and *future_guess_set.* At the point of splitting *prev_guess_index* (i.e., λ_{g1}) of PROB is assigned to *selected_wavelength* of both RES_FWD and RES_BKD. The RES_BKD moves towards the *source*, reserving λ_{g1} (i.e. the wavelength stored at *selected_wavelength*) and deleting the entries of this request in *node_tables* on the way. The RES_FWD moves towards *destination* reserving $\lambda_{g1.}$

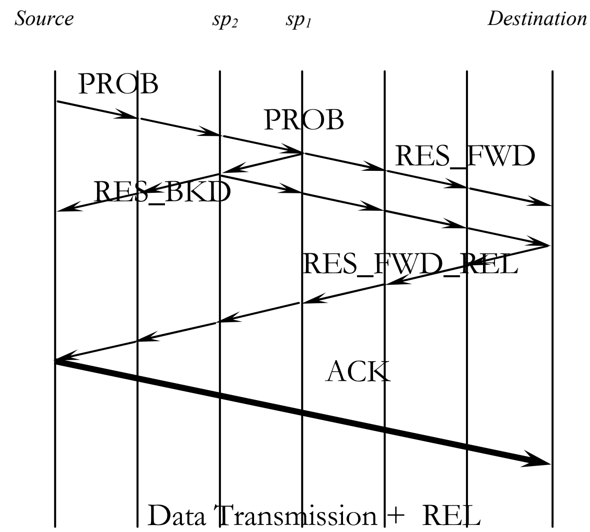

Figure 8. Case of success in MMSRP.

However, if RES_BKD fails at some *intermediate node* due to non-availability of $\lambda_{g1,}$ further splitting may occur (maximum *b*-1 times). In that case, the node selects next candidate from the *future_guess_set*, subject to availability both in *present_link* and *previous_link*. Then RES_BKD again splits into two new reservation packets RES_FWD and RES_BKD. These RES_FWD and RES_BKD, act like earlier RES_FWD and RES_BKD packets respectively. Both packets attempt to reserve the selected wavelength in the same way, in both forward and backward directions. RES_FWD on its way also releases the previously reserved wavelengths by previous RES_FWD and RES_BKD.

If both RES_FWD and RES_BKD are successful to reserve the same wavelength, data transmission starts after receiving the ACK from *destination*. If RES_BKD is stuck at an *intermediate node* and all possible splittings are exhausted, the request is blocked and RES_BKD is converted into NACK which moves towards *source*. Another REL_FWD is generated from the point of failure which moves towards *destination* and releases the wavelengths reserved so far by both RES_FWD and RES_BKD. Again, if RES_FWD fails, it is converted into REL_BKD which moves towards *source* releasing the wavelengths reserved so far. It also acts as a NACK and deletes the entries of this request in *node_tables* of the path. Figure 8 shows a case of success whereas Figure 9 shows a case of failure. In the figures, sp_1 and sp_2 indicate the two *splitting points* (nodes) of a request.

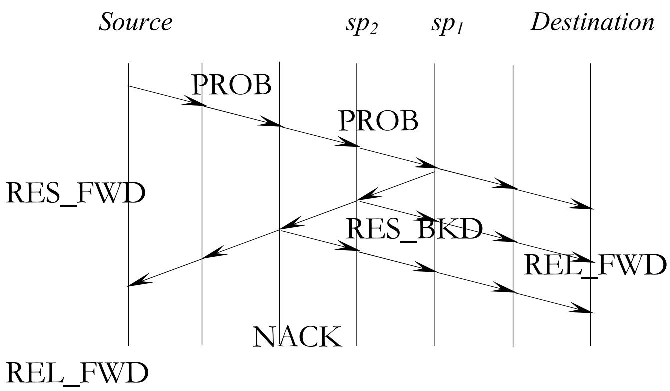

Figure 9. Case of failure in MMSRP.

We have studied MMSRP exhaustively and compared with its peers. Some representative results are presented here. Fig.10 represents the variation of *bp* with *cr* for *wl*=200. In general, as expected, for all the schemes, *bp* increases with increase in *cr*. From the figure, we find that MMSRP always performs better than MSRP and MBRP. Also we can observe that, relative performance is much better for high values of *cr* (i.e., at high load). It happens because, at high load, the crisis of getting a wavelength is more, and the failure cases use the *future_guess_set* of MMSRP, leading to more successful cases.

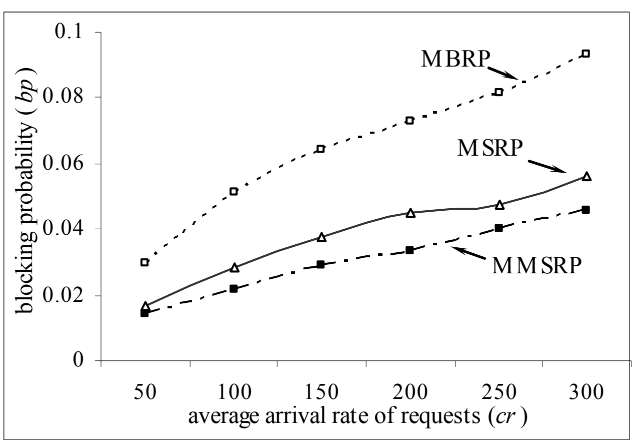

Fig. 10: Variation of *bp* with *cr* for fixed *wl*=200

VI. COMPARISON OF PROTOCOLS

In the previous sections we have discussed different schemes, where *multiple wavelengths* are used for lightpath establishment. The protocols are separately compared with their peers and their performances are found to be quite promising. In this section we have presented comparative study of these three protocols, DIMRP, SRP and MMSRP.

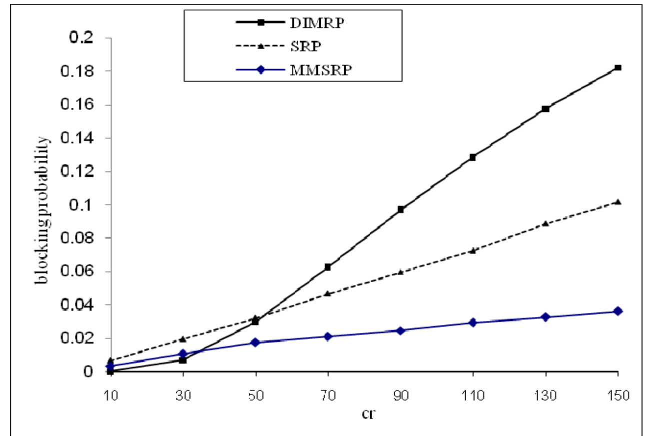

Fig.11. Variation of *bp* with *cr* for *wl* =50.

Since the key performance parameter in lightpath establishment is *bp*, we have studied *bp* exhaustively. The proposed scheme, MMSRP is compared with SRP and DIMRP. One representative result for *wl*= 200 is shown in Fig. 11. From the figure, in general it is found that for all the protocols, *bp* increases with increase in *cr*. However MMSRP performs distinctly better than other two. Also it can be observed from the Fig. 11 that, with the increment of *cr* the relative performance of MMSRP also improves. This happens because, as *cr* increases, crisis also increases and a reasonable amount of blocking happens due to collision of wavelength, when more than one requests attempt to capture same wavelength even if other free wavelengths are present. Random selection method, used in DIMRP and SRP, cannot prevent this. In MMSRP, probabilistic method of selection of wavelength takes care of the wavelength collision and utilizes available wavelengths more efficiently. Another important thing is that, MMSRP uses the *future_guess_set*, through the process of *multiple splitting*, and successfully reduces *bp.*

VII. CONCLUSION

MMSRP using multiple splitting, combined with *multi-wavelength* guessing, reduces *blocking probability* considerably, compared to SRP and DIMRP. During probing, first splitting is used dynamically and then *multiple splitting* is used in case of failures. Thus, MMSRP performs better than the current best protocols as far as *blocking probability* is concerned at higher wavelength regions. So it may be considered as a better performer in DWDM networks, specially for the applications, where protocol efficiency is of prime importance and the network uses a larger number of wavelengths.

REFERENCES

[1] H. Zang, J. P. Jue and B. Mukherjee, "Review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks", *Optical networks*, Vol 1, no 1, pp 47-60, Jan. 2000.

[2] D. Saha, "A Comparative study of distributed protocols for wavelength reservation in WDM optical networks", SPIE Opt. Netw. Mag., vol 3, no. 1, pp. 45-52, 2002.
[3] K. Chan and T. P. Yum, "Analysis of least congested path routing in WDM lightwave networks," in *Proc. IEEE INFOCOM'*1994, pp. 962–965.
[4] H Zang, J. P. Jue., L Sahashrabuddhe, R Ramamurthy, and B Mukherjee," Dynamic Lightpath Establishment in Wavelength-Routed WDM Networks", IEEE Commun Magazine, pp 100-108, Sept. 2001.
[5] C. S. Murthy and M. Gurusamy, *WDM Optical Networks, Concepts, Design and Algorithms*. Englewood Cliffs, NJ: Prentice-Hall, 2001.
[6] K. Lu, J. P. Jue and G. Xiao, "Intermediate-Node Initiated Reservation (IIR): A New Signaling Scheme for Wavelength-Routed Networks," IEEE JSAC, vol. 21, no. 8, pp 1285-1294, 2003.
[7] W. Lin, R. S. Wolff and B. Mumey, "A Markov-Based Reservation Algorithm for Wavelength Assignment in All-Optical Networks", IEEE Journal of Light wave Technologies, vol. 25, no. 7, pp 1676-1683, 2007.
[8] M. Sengupta, S. K. Mondal and D. Saha "An Adaptive Split Reservation Protocol (SRP) for Dynamicaly Reserving Wavelengths in WDM Optical Networks", Proc. ICDCN 2008, LNCS 4904, pp 440-451, 2008.
[9] M. Sengupta, S. K. Mondal, C. Bose and D. Saha, "A Markov selection Split Reservation Protocol for WDM Optical Networks without Wavelength Conversion", Proc. . *IEEE GLOBECOM, 2008.*
[18]
[10] P. Lertchusakul, X. Wang, H. Morikawa and T. Aoyama, "Priority based Distributed Protocol for Wavelength Reservation in WDM optical networks", April 2003.
[11] A.E. Ozdaglar and D.P.Bertsekas, "Routing and Wavelength Assignment in Optical Networks", IEEE/ACM Trans. On Networking, vol. 11, no. 2, pp. 259-271, April. 2003.
[12] T. Ozugur, M. Park, and J. Jue, "Label prioritization in GMPLS-centric all-optical networks," in *Proc.ICC*, 2003, pp. 1283–1287.
[13] M. Sengupta, S. K. Mondal and D. Saha "Destination Initiated Multi-wavelength Reservation Protocol (DIMRP) in WDM Optical Networks: Finding the Optimal Selectivity for Wavelength Assignment", *Proc..ICDCN*, pp. 497-502, 2006.
[14] M. Sengupta, S. K. Mondal, and D. Saha, "MMSRP: Multi-Wavelength Markov-based Split Reservation Protocol for DWDM Optical Networks**",** *Proc. The 2010 ACM Symposium on Applied Computing*.
[15] A. Sridharan and K. N. Sivarajan, "Blocking in All-Optical Networks," *Proc. IEEE INFOCOM* , pp. 990-999, 2000.
[16] B. Mukherjee, "WDM optical communication networks: progress and challenges," *IEEE Journal on Selected Areas in Communications*, vol. 18, no.10, pp. 1810-1824, October, 2000.
[17] R. Ramaswami and K. N. Sivarajan. "Routing and Wavelength Assignment in All-Optical Networks," *IEEE/ACM Transactions on Networking*. vol. 3, pp. 489-500. October 1995.