



<b>Title</b>	<b>Active routing service for the next-generation network/ISDN3</b>
<b>Author(s)</b>	<b>Lam, RYW; Chan, HCB; Li, VOK; Dillon, TS; Leung, VCM</b>
<b>Citation</b>	<b>Conference Record / IEEE Global Telecommunications Conference, 2002, v. 3, p. 2375-2379</b>
<b>Issued Date</b>	<b>2002</b>
<b>URL</b>	<b><a href="http://hdl.handle.net/10722/46374">http://hdl.handle.net/10722/46374</a></b>
<b>Rights</b>	<b>Creative Commons: Attribution 3.0 Hong Kong License</b>

# Active routing service for the next-generation network/ISDN3

Ray Y. W. Lam<sup>1</sup>, Henry C.B. Chan<sup>1</sup>, *Member, IEEE*, Victor O. K. Li<sup>2</sup>, *Fellow, IEEE*,  
Tharam S. Dillon<sup>3</sup>, *Fellow, IEEE*, and Victor C. M. Leung<sup>4</sup>, *Senior Member, IEEE*

<sup>1</sup>Department of Computing, Hong Kong Polytechnic University, Hong Kong

<sup>2</sup>Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong

<sup>3</sup>Department of Computer Science and Computer Engineering, La Trobe University, Australia

<sup>4</sup>Department of Electrical and Computer Engineering, University of British Columbia, Canada

**Abstract** - In recent years, a new routing method, known as active routing, has been emerging. This involves using active packets to configure customized network paths. Based on a Markov decision model, this paper presents an active routing service for active networks in general and the next generation network, called ISDN3, in particular. Our aim is to determine the active routing policy so as to minimize the network cost. Theoretical analysis is presented to show the advantages of our proposal as compared with three other approaches.

## I. INTRODUCTION

Generally speaking, routing is a key network function for forwarding packets over a network in an efficient manner [1]. In connection-oriented networks (e.g., Asynchronous Transfer Mode (ATM)), communication paths are usually fixed or reserved during call set-up in accordance with the user requirements. In connectionless networks (e.g., the Internet), packets are forwarded by routers on a hop-by-hop basis [2]. The routing table in each router, and thus the communication paths, can be changed dynamically based on the traffic. In the Internet, two types of routing protocols are commonly used to support intra-domain routing: the distance vector (e.g., Routing Information Protocol (RIP) [3]) and link state (e.g., Open Shortest Path First (OSPF) [4]) routing protocols. For inter-domain routing, policy-based routing protocols, such as Border Gateway Protocol (BGP) [5], are often used. Recently, there has been considerable interest in developing quality of service (QoS) routing protocols, which can forward packets based on a particular QoS requirement [6]. From the end users' point of view, the above routing protocols are "passive" because the end users have very little control over the routing process, other than providing the essential information (e.g., destination address). With the advent of active networks [7, 8], a new routing method, known as active routing [9, 10], has become available. Basically, it enables an end-user application to specify its routing policy. By using active routing, customized and flexible routing services can be provided. The combination of passive and active routing services will open a new era of innovative services in next generation high-speed networks. Although each user can employ his/her own active routing algorithm, this paper formulates a Markov decision problem to support active routing so that the consumption of network

resources can be minimized. The organization of the rest of the paper is as follows. Section II gives an overview of ISDN3 and active routing. Section III presents a Markov decision model to support active routing. Section IV analyses the performance of active routing as compared with other approaches. Section V gives the conclusions.

## II. OVERVIEW OF ISDN3 AND ACTIVE ROUTING

In general, active routing can be used in any active network. Here, we consider using it in the next generation network, called ISDN3, as proposed in [11]. In our view, the convergence of ATM, the Internet and active networks will form the basis for ISDN3. In ISDN3, not only network traffic, but also network functions are integrated, thus creating a truly integrated network. To support ISDN3, a new type of packet, called the quantum packet, is proposed. Basically, a quantum packet consists of one or more quanta, which are forwarded together by the network. Effectively, a quantum packet can be viewed as an extended ATM cell. Each quantum carries a quantum label to indicate the type of network function required (e.g., switching/routing/active). An integrated traffic-forwarding device, called the Forwarding Engine (FEE), is used to forward different types of quantum packets in a unified manner. One type of quantum packet is active and programmable. It can be used to support the active routing service in ISDN3.

In active routing, an end-user application tells the network how to forward its packets. To do this, a Network Object Model (NOM) is required to allow the active packets to manipulate the network information. This is similar to how JavaScript manipulates a Web browser using the well-known Document Object Model [12]. Basically, the NOM provides a tree-like interface for active packets to access the network information used to make the routing decision. Here, we assume that the network cost, and the associated path for sending packets from one node to any of the other nodes, can be retrieved from the NOM. Such information is updated regularly, at the beginning of each routing cycle. At the beginning of a communication session, either the sender's or receiver's process sends an active packet to read the cost information from the NOM. It then interacts with the active network module of the respective FEEs to set up the forwarding tables along the associated path. Once the forwarding tables are set up, subsequent packets can be forwarded by means of cut-through switching. At the beginning of each routing cycle, the end-user application can send an active packet to determine whether there is a better

Acknowledgements: The work described in this paper has been supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China under account number PolyU 5085/00E. The paper has been refined with the assistance of Mr. Peter Meldrum.

path according to its policy. Should a better path be available, it can reconfigure the path and remove the old one. In this paper, we formulate a Markov decision problem to determine the routing policy so as to minimize the average network cost for a communication session.

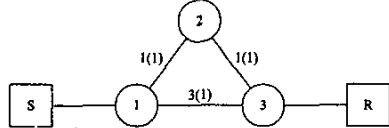


Fig. 1. A network with two paths where the link weights are the link costs in the 1<sup>st</sup> cycle and the 5<sup>th</sup> cycle (inside the brackets).

### III. MARKOV DECISION MODEL FOR ACTIVE ROUTING

This section presents a Markov decision model to support active routing. Suppose that a discrete time system is used and the network cost is updated at time 1, 2, 3, and so on. We denote the time interval between  $t$  and  $t + 1$  as the  $t^{\text{th}}$  routing cycle. Consider that a sender sets up a communication session of duration  $d$  time units with a receiver (i.e., the session starts at time 1 and ends at time  $d + 1$ ). Therefore, the session has  $d$  routing cycles. At the beginning of the session, the sender's active packet chooses the minimum cost path and then reserves it for the subsequent routing cycles. Once the path is reserved, the cost is fixed until changes occur in the path. This means that the sender can at least use the existing path at the current reserved cost for the subsequent cycles. Later, a lower cost path may become available. A sender's active packet may switch to the better path by paying a one-time switching cost,  $C_{\text{switch}}$ . After switching, the newly reserved cost applies to the current and subsequent cycles until switching occurs again. The cost for each cycle is determined by adding the current reserved cost and any switching cost. The total cost for the whole session is the sum of the costs for all  $d$  cycles. The objective of our active routing policy is to minimize the total cost for a session. Fig. 1 shows a simple example to illustrate the problem. Suppose that a sender S and a receiver R are connected by two paths:  $1 \rightarrow 2 \rightarrow 3$  and  $1 \rightarrow 3$ . At the 1<sup>st</sup> cycle (session setup), the path  $1 \rightarrow 2 \rightarrow 3$  is reserved because it has a smaller cost of  $1 + 1 = 2$ . At the 5<sup>th</sup> cycle, the cost of the path  $1 \rightarrow 3$  becomes 1. Since a lower cost path is now available, the session needs to decide whether to keep the path  $1 \rightarrow 2 \rightarrow 3$  or switch to the path  $1 \rightarrow 3$  by paying a switching cost, say 10. If the session is going to terminate after two further cycles, and the subsequent network costs remain unchanged, then the cost of the remaining cycles will be  $10 + 1 + 1 = 12$  if switching is performed, or  $2 + 2 = 4$  otherwise. In this case, it is preferred not to switch, even though a lower-cost path is available. Note that, the actual problem is more complicated than this because the network cost is probabilistic.

Following the notation in [13, 14], we formulate a Markov decision model as follows. We assume that the network cost  $c_j$  is governed by a probability distribution  $N(c_j)$ , where  $c_j > 0$ ,  $j = 1, 2, \dots, n$ , and  $c_j < c_{j+1}$  for all  $j$ . Decisions are made at the beginning of each routing cycle. The set of decision epochs is denoted by  $T = \{1, 2, \dots, d\}$ . We define  $(c, s)$  as the system

state, where  $c$  is the current reserved cost and  $s$  is the available cost saving. If  $s > 0$ , the current network cost is  $c - s$ , whereas if  $s = 0$  it means that the current network cost is larger than or equal to the current reserved cost. There are  $n(n + 1)/2$  possible states at each decision epoch. The set of states at  $t$  is denoted by  $S_t$  as follows:

$$S_t = \{c_i, (c_i - c_j)^+\}, \quad (1)$$

where  $i = 1, 2, \dots, n$  and  $j = 1, 2, \dots, i$ , for  $t = 1, 2, \dots, d + 1$ .

Note that,  $S_{d+1}$  is the set of terminal states (i.e., when the session is terminated). Let  $A_{(c,s),t}$  denotes a set of actions that are to be chosen at decision epoch  $t$  with state  $(c, s) \in S_t$ :

$$A_{(c,s),t} = \begin{cases} \{1 (\text{Switch}), 0 (\text{Do not switch})\}, & \text{if } s > 0 \\ \{0\}, & \text{if } s = 0 \end{cases} \quad (2)$$

where  $t = 1, 2, \dots, d$ . Obviously, it is not cost-effective to switch if  $s = 0$  (i.e., there is no cost saving).

Let  $Cost_t(c, s, a)$  denote the cost for the  $t^{\text{th}}$  routing cycle if the system state is  $(c, s) \in S_t$ , and the selected action is  $a \in A_{(c,s),t}$  at  $t$ . It is defined by the following:

$$Cost_t(c, s, a) = \begin{cases} c, & \text{if } a = 0 \\ c - s + C_{\text{switch}}, & \text{if } a = 1 \end{cases}, \quad t = 1, 2, \dots, d \quad (3)$$

If the choice is to "Switch", the cost of the current cycle is the new reserved cost plus the switching cost. If the choice is "Do not switch", the cost is equal to the existing reserved cost.

Let  $P((k, l)|(i, j), a)$  denotes the state transition probability from the current state,  $(i, j) \in S_t$ , at  $t$ , to the next state,  $(k, l) \in S_{t+1}$  at  $t + 1$ , with the selected action  $a \in A_{(i,j),t}$ . It is given by expression (4) for  $t = 1, 2, \dots, d$ .

$$P((k, l)|(i, j), a) = \begin{cases} \sum_{c_2 \geq k} N(c_2), & \text{if } k = i, l = 0 \text{ and } a = 0 \text{ or} \\ & \text{if } k = i - j, l = 0 \text{ and } a = 1 \\ N(k - l), & \text{if } k = i, l > 0 \text{ and } a = 0 \text{ or} \\ 0, & \text{if } k = i - j, l > 0 \text{ and } a = 1 \\ & \text{otherwise} \end{cases} \quad (4)$$

If the action is to "Switch", the reserved cost for the next state (i.e.,  $k$ ) becomes  $i - j$ ; otherwise, it remains unchanged. The first term in expression (4) represents the transitions with no cost savings (i.e.,  $l = 0$ ). The corresponding probability is the sum of  $N(c_2)$  for all  $c_2 \geq k$ . The second term represents the cases with cost savings.

Let  $v^t(c, s)$  denote the minimum expected accumulated cost for routing cycles  $t, t + 1, \dots, d + 1$ , when the system state is  $(c, s) \in S_t$  at  $t$ . According to [13],  $v^t(c, s)$  and  $v^{t+1}(i, j)$  are related by a recursive equation for  $t = 1, 2, \dots, d$ , as follows:

$$v^t(c, s) = \min_{a \in A_{(c,s),t}} \left\{ Cost_t(c, s, a) + \sum_{(i,j) \in S_{t+1}} P((i, j)|(c, s), a) v^{t+1}(i, j) \right\} \quad (5)$$

To solve the recursive equation, the expected cost for each terminal state  $(c, s) \in S_{d+1}$  is initialized as

$$v^{d+1}(c, s) = 0. \quad (6)$$

Starting from  $t = d + 1$ , we can calculate recursively  $v^t(c, s)$  for each state  $(c, s) \in S_t$  at  $t = 1, 2, \dots, d$ .

At each decision epoch  $t$ , the sender's active packet needs to select an action  $a \in A_{(c,s),t}$  to make  $v^t(c, s)$  as small as

possible. Let  $A_{(c,s),t}^*$  be the preferred action for state  $(c, s)$  at  $t$ . This means that it is the action with the smallest  $v^t(c, s)$ . By using the backward induction algorithm, we can obtain the minimal expected cost for the whole session (i.e.,  $v^1(c, s)$ ) for each initial state  $(c, s) \in S_1$ . Furthermore, the set of preferred actions for different states at each decision epoch (i.e., the routing policy) can be found.

Let us look at a simple example (see [14] for more examples on Markov decision problems). Suppose that the network costs are 1, 2 and 3 with equal probability. Therefore,  $c_j = j$  and  $N(c_j) = 1/3$ , for  $j = 1, 2$  and 3. The set of states are  $S_t = \{(1, 0), (2, 0), (2, 1), (3, 0), (3, 1), (3, 2)\}$ , for  $t = 1, 2, \dots, d + 1$ . When the system state is  $(3, 1)$  and the chosen action is to "Switch", the probability that the next state is  $(2, 0)$  is  $N(c_2) + N(c_3) = 2/3$ . Assuming that  $C_{switch} = 2$ , the corresponding value of  $Cost_2(3, 1)$  is  $3 - 1 + 2 = 4$ . Other transitional probabilities and  $Cost_2(c, s, a)$  can be calculated similarly.

Let the session duration be two routing cycles. We define  $v^t(c, s, a)$  as follows:

$$v^t(c, s, a) = Cost_2(c, s, a) + \sum_{(i,j) \in S_{t+1}} P((i, j) | (c, s), a) v^{t+1}(i, j) \quad (7)$$

Here, we demonstrate how backward induction is carried out at each decision epoch  $t$ :

Step 1. Initialize  $t = 3$  and  $v^3(c, s) = 0$ , for all  $(c, s) \in S_3$ .

Step 2. For  $t = 2$ , compute

$$v^2(2, 1, 0) = Cost_2(2, 1, 0) + P((2, 0) | (2, 1), 0) v^3(2, 0) + P((2, 1) | (2, 1), 0) v^3(2, 1) = 2$$

$$v^2(2, 1, 1) = Cost_2(2, 1, 1) + P((1, 0) | (2, 1), 1) v^3(1, 0) = 3$$

Therefore,  $v^2(2, 1) = 2$  and  $A_{(2,1),2}^* = 0$ . Table I gives  $v^2(c, s, a)$ ,  $v^2(c, s)$  and  $A_{(c,s),2}^*$  for each state  $(c, s) \in S_2$ . The impossible actions are denoted by "N".

TABLE I SUMMARY AT THE FIRST (IN BRACKETS {}) AND SECOND DECISION EPOCHS

		(1, 0)	(2, 0)	(2, 1)	(3, 0)	(3, 1)	(3, 2)
$v^2(c, s, a)$	$a = 0$	1{2}	2{4}	2{4}	3{6}	3{6}	3{6}
	$a = 1$	N	N	3{4}	N	4{6}	3{4}
$v^2(c, s)$	$\{v^2(c, s)\}$	1{2}	2{4}	2{4}	3{6}	3{6}	3{4}
$A_{(c,s),2}^*$	$\{A_{(c,s),2}^*\}$	0{0}	0{0}	0{0}	0{0}	0{0}	0{1}

Step 3. Finally compute  $v^1(c, s, a)$ ,  $v^1(c, s)$  and  $A_{(c,s),1}^*$  for each state  $(c, s) \in S_1$  as given by the terms in brackets in Table I.

As shown above, the only worthwhile case for switching is when  $t = 1$ , and the current reserved cost is 3 and the current network cost is 1 ( $s = 2$ ). In most cases, switching is not preferred, even when a lower cost is available, because of the short session duration and the high switching cost.

#### IV. ANALYSIS

In this section, we apply the above Markov decision model to analyze the active routing policy. We assume that the network cost  $c_j$  is normally distributed with mean  $\mu$  and standard deviation  $\sigma$ . More specifically, we assume that

$$c_j = \mu - \frac{(n-1)\delta\sigma}{2} + (j-1)\delta\sigma \quad j = 1, 2, \dots, n. \quad (8)$$

$N(c_j)$  is given by the area under the respective normal curve, which is bounded by  $c_j \pm \Delta$  where  $\Delta$  equals a half interval (i.e.,  $\delta\sigma/2$ ). For the leftmost ( $c_1$ ) and rightmost ( $c_n$ ) cost levels, the areas are extended to  $-\infty$  and  $\infty$ , respectively.

In the analysis,  $n = 51$  and  $\delta = 0.2$ . Our aim is to evaluate the average cost  $C$  for a session under the following schemes: **Fixed (FIX)**: The network cost is reserved, at the time the session is set-up, and is then fixed for the whole session. This resembles the approach used by the connection-oriented services (e.g., ATM).

**Dynamic without cost reservation (DYN)**: The cost of each cycle is changed according to the updated network cost. This resembles the approach used by the connectionless services as communication paths are changed dynamically.

**Dynamic with cost reservation (DCR)**: The network cost is reserved in this scheme. Furthermore, if the current network cost is smaller than the reserved network cost, the latter will be updated to the former.

**Active routing (ACT)**: The decision policy, as described in Section III, is used to support active routing.

For FIX, the total average cost can be easily found to be

$$C = d\mu. \quad (9)$$

For DYN, the switching cost  $C_{switch}$  is added to the network cost in each routing cycle. Hence, the total average cost is

$$C = d\mu + dC_{switch}. \quad (10)$$

For ACT, by using the discrete costs  $c_j$  and the associated probabilities  $N(c_j)$ , we can formulate the states and compute the transition probability matrix. For a given  $d$ , we can then determine the preferred actions by using the backward induction algorithm. Finally, the total cost is given by

$$C = \sum_{j=1}^n v^1(c_j, 0) N(c_j). \quad (11)$$

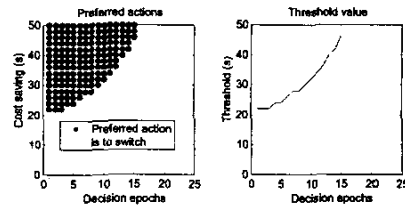


Fig. 2. a (left) Preferred actions for different values of cost saving, and b (right) the corresponding threshold values at different decision epochs.

Referring to our earlier numerical example, the total average cost is  $2(1/3) + 4(1/3) + 6(1/3) = 4$ . Besides the total cost, it is interesting to study how different parameters could affect the decision policy. Fig. 2a shows the preferred actions  $A_{(c,s),t}^*$  for states  $(c, s) = (250, 0), (250, 2), \dots, (250, 50)$  at  $t = 1, 2, \dots, 25$ , when  $\mu = 250$ ,  $\sigma = 10$ ,  $d = 25$  and  $C_{switch} = 500$ . Each of the points in Fig. 2a shows that the preferred action is to "Switch". It can be seen that the policy has the following form: at a particular  $t$ , switching is preferred if the cost saving reaches a certain threshold. For example, at  $t = 14$ , it is preferred to switch if the cost saving is greater than or equal to 42. Otherwise, no switching is preferred. Fig. 2b shows

the corresponding threshold values for different decision epochs when the reserved cost is equal to  $\mu$ .

For DCR, we use a slightly modified backward induction method to calculate the total average cost. In this scheme, when the cost saving is greater than 0, switching will be performed. Mathematically,  $v(c, s)$  is expressed as follows:

$$v(c, s) = \begin{cases} Cost_t(c, s, 1) + \sum_{(i, j) \in S_{t,1}} P((i, j) | (c, s), 1) v^{t+1}(i, j), & \text{if } s > 0 \\ Cost_t(c, s, 0) + \sum_{(i, j) \in S_{t,0}} P((i, j) | (c, s), 0) v^{t+1}(i, j), & \text{if } s = 0 \end{cases} \quad (12)$$

, for  $t = 1, 2, \dots, d$ .

Based on (6) and (12), the total average cost can be obtained by using (11).

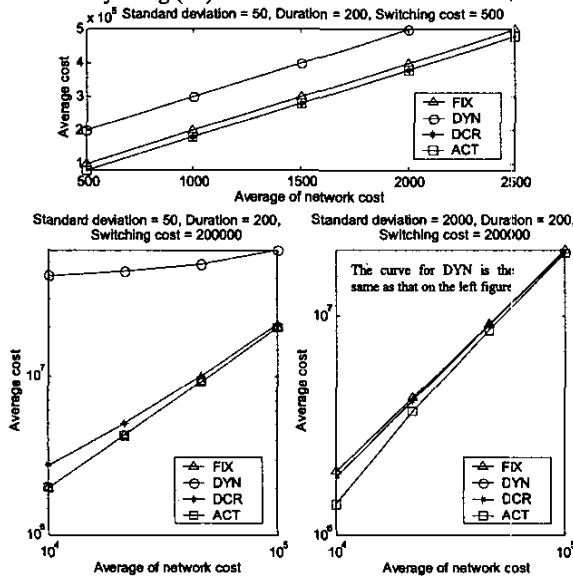


Fig. 3. Average cost vs. the average of the network cost.

To carry out the analysis and comparison, we set our base parameters to the following:  $\mu = 10,000$ ,  $\sigma = 2,000$ ,  $d = 200$  and  $C_{switch} = 200,000$ . These parameters are varied in turn to study their effects on the average cost and threshold values for the active routing approach. Fig. 3 shows the average cost when  $\mu$  is varied. Recall that the average cost at each routing cycle is composed of the reserved cost and the switching cost (if any). From the top figure, it can be seen that the average cost for all schemes is directly proportional to  $\mu$  with a slope equal to  $d$ . In FIX and DYN, this can be seen easily from (9) and (10), respectively. In DCR and ACT, varying  $\mu$  does not alter the switching policy. Therefore, the total switching cost is constant even when  $\mu$  is changed. For the same reason, the threshold values at different decision epochs are the same irrespective of the value of  $\mu$ . From the figure, it can be seen that DCR and ACT are more favorable when  $\sigma$  and  $C_{switch}$  are both small. However, as  $C_{switch}$  increases, the average cost for DCR will also increase, making it higher than that of FIX and ACT. ACT always gives the best performance, particularly when both  $\sigma$  and  $C_{switch}$  are large.

Fig. 4 shows the effect on the average cost when the standard deviation  $\sigma$  is varied. The average cost for FIX and DYN is independent of  $\sigma$ . The figure shows that the average cost for FIX, DCR and ACT is comparable when  $\sigma$  and  $C_{switch}$  are both small. As  $\sigma$  increases, the average cost decreases in DCR and ACT. When  $C_{switch}$  is small, DCR and ACT have almost the same average cost. Again, the average cost increases significantly in DYN and DCR as  $C_{switch}$  increases. However, the average cost remains at a low value in ACT even when  $C_{switch}$  is large. Furthermore, DCR has a lower average cost than FIX when  $\sigma$  and  $C_{switch}$  are both large.

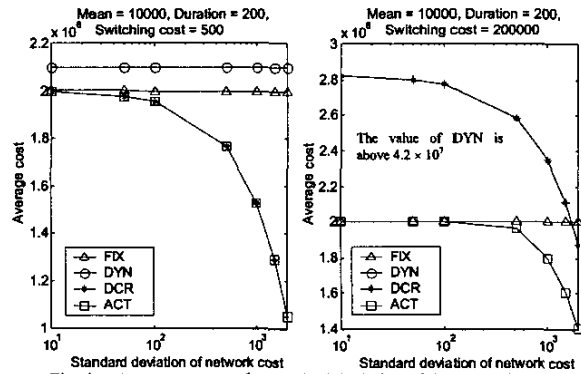


Fig. 4. Average cost vs. the standard deviation of the network cost.

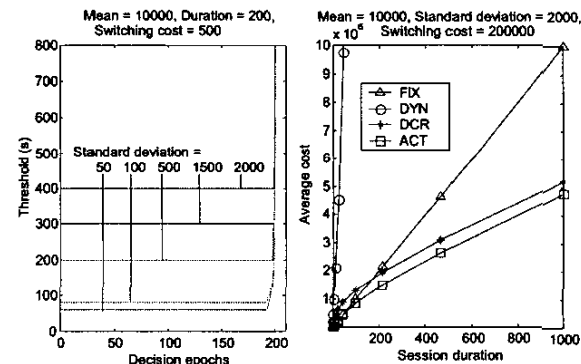


Fig. 5. Threshold values for different values of the standard deviation.

Fig. 6. Average cost vs. the duration of the transmission.

Fig. 5 shows the threshold values for different values of  $\sigma$ . It can be seen that the nearer the session comes to an end, the larger the values are of the threshold. Towards the end of a session (about 10 routing cycles from the end), the threshold values for different values of  $\sigma$  are very close. Since switching is expensive, the active routing approach prefers not to switch unless the cost saving is very large. In comparison, the threshold values are dramatically lower and maintain stable values when the session is further from the end. The stable threshold values increase as  $\sigma$  increases.

Fig. 6 shows the average cost when the duration of the session  $d$  is varied. When  $d$  is small, the average cost is close for all schemes. As  $d$  increases, the average cost increases at

a different rate for each scheme. The average cost for DYN is the most sensitive to an increase in  $d$ , whereas DCR and ACT are the least sensitive. This is because, in DCR and ACT, the longer the duration of a session, the higher the chance is that a lower network cost can be reserved.

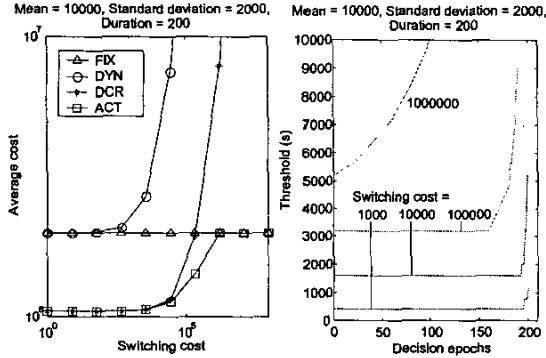


Fig. 7. Average cost vs. the switching cost.

Fig. 8. Threshold values for different values of the switching cost.

Fig. 7 shows the average cost when the switching cost  $C_{switch}$  is varied. When  $C_{switch}$  is small, the average cost is close in FIX and DYN and in DCR and ACT. Since no switching is required, the average cost for FIX is independent of  $C_{switch}$ . The average cost for DYN is the most sensitive to a change in  $C_{switch}$  because switching occurs in every routing cycle. In DCR, switching occurs when a lower network cost is available. Therefore, the average cost for DCR increases less dramatically than it does in DYN, as  $C_{switch}$  increases. For ACT, the average cost is even less sensitive to an increase in  $C_{switch}$ . When  $C_{switch}$  is small, switching occurs as frequently as it does in DCR. When  $C_{switch}$  is large, active routing prefers not to switch. Therefore, it performs no worse than FIX even when  $C_{switch}$  is very large. This contrasts with DYN and DCR in which the average cost rises far above that of FIX. From the above results, we can see clearly that active routing adapts well to different situations and always gives the minimum average cost.

Fig. 8 shows the threshold values in ACT for different values of  $C_{switch}$ . It can be seen that as  $C_{switch}$  increases by an order of magnitude, the stable threshold values increase steadily. Towards the end of a session, the active routing approach prefers not to switch, particularly when  $C_{switch}$  is large, unless the cost saving is very large. Table II summarizes the effects of different parameters on the performance. A change in  $\sigma$  produces a more significant effect, than a change in  $C_{switch}$ , on both the average cost and the threshold values in ACT. Interestingly, when  $\sigma$  is increased tenfold, it is found that the decision policy is similar to that when  $C_{switch}$  is increased tenfold, except that the threshold values increase tenfold as well. The active routing approach provides the optimal performance in all situations. In particular, it can reduce the average cost significantly, as compared with other schemes, when  $\sigma$ ,  $C_{switch}$  and  $d$  are all large. For example, using the base parameters, it

can decrease the average cost for a session by about 30% and 25% as compared with that in FIX and DCR, respectively. The cost saving as compared with DYN is even more significant due to the high switching cost.

TABLE II. SUMMARY OF THE ANALYSIS

Increase in	Effect on expected total cost $C$				Effect on threshold	Preferred schemes
	FIX	DYN	DCR	ACT		
$\mu$	↑	↑	↑	↑	=	NA
$\sigma$	=	=	↓1	↓1	↑	DCR / ACT
$d$	↑2	↑1	↑3	↑3	=	DCR / ACT
$C_{switch}$	=	↑1	↑2	↑3	↑	FIX / ACT

Key:

↑: increase the value ↓: decrease the value =: no effect on the value  
 The number indicates the degree of effect: 1: most affected 3: least affected  
 \*Threshold remains constant unless  $d$  is very small.

## V. CONCLUSION

In conclusion, we have presented an active routing service for active networks and next generation networks. This involves using active packets to configure customized data forwarding paths based on the network information. With the aim of minimizing network cost, a Markov decision model has been formulated to support active routing. According to the model, the paths should be reconfigured if a particular cost saving threshold is reached. The threshold values, which depend on the network parameters, can be found by solving the Markov decision model. Theoretical analysis confirms that by using the proposed routing policy, the average network cost can be minimized under various conditions as compared with other approaches.

## REFERENCES

- [1] M. Steenstrup, *Routing in communications networks*, Prentice Hall, 1995.
- [2] W.R. Stevens, *TCP/IP illustrated volume 1*, Addison Wesley Longman, 1994.
- [3] G. Malkin, "RIP version 2," *Internet RFCs 2453*, Nov. 1998.
- [4] J. Moy, "OSPF version 2," *Internet RFCs 2328*, Apr. 1998.
- [5] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)," *Internet RFCs 1771*, Mar. 1995.
- [6] S. Chen and K. Nahrstedt, "An overview of quality of service routing for next-generation high-speed networks: problems and solutions," *IEEE Network*, vol. 12, no. 6, pp. 64-79 Nov./Dec. 1998.
- [7] K.L. Calvert, S. Bhattacharjee, E. Zegura, and J. Sterbenz, "Directions in active networks," *IEEE Comm. Mag.*, vol. 36, no. 10, pp. 72-78, Oct. 1998.
- [8] D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie, D.J. Wetherall, and G.J. Minden, "A survey of active network research," *IEEE Comm. Mag.*, vol. 35, no. 1, pp. 80-86, Jan. 1997.
- [9] N.F. Maxemchuk and S.H. Low, "Active routing," *IEEE Journal on Sel. Areas in Comm.*, vol. 19, no. 3, pp. 552-565, Mar. 2001.
- [10] C. Tschudin, H. Lundgren and H. Gulbrandsen, "Active routing for ad hoc networks," *IEEE Comm. Mag.*, vol. 38, no. 4, pp. 122-127, Apr. 2000.
- [11] H.C.B. Chan, R.Y.W. Lam, T.S. Dillon, V.O.K. Li, and V.C.M. Leung, "ISDN3: The next generation networks," *Proc. IEEE PACRIM'01*, vol. 2, pp. 607-610, Victoria, B.C., Canada, 2001.
- [12] T.R. Nieto, H.M. Deitel, P.J. Deitel, H. Deitel, and P. Deitel, *Complete Internet and World Wide Web programming training course*, Prentice Hall, 2000.
- [13] M.L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*, 1<sup>st</sup> edition, Wiley-Interscience, 1994.
- [14] M.L. Puterman, "Dynamic programming," *Encyclopedia of physical science and technology*, vol. 4, Academic Press Inc., pp. 438-463, 1987.