<table>
<thead>
<tr>
<th><strong>Title</strong></th>
<th>Active routing service for the next-generation network/ISDN3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Author(s)</strong></td>
<td>Lam, RYW; Chan, HCB; Li, VOK; Dillon, TS; Leung, VCM</td>
</tr>
<tr>
<td><strong>Citation</strong></td>
<td>Conference Record / IEEE Global Telecommunications Conference, 2002, v. 3, p. 2375-2379</td>
</tr>
<tr>
<td><strong>Issued Date</strong></td>
<td>2002</td>
</tr>
<tr>
<td><strong>URL</strong></td>
<td><a href="http://hdl.handle.net/10722/46374">http://hdl.handle.net/10722/46374</a></td>
</tr>
<tr>
<td><strong>Rights</strong></td>
<td>This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.; ©2002 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.</td>
</tr>
</tbody>
</table>
Active routing service for the next-generation network/ISDN3

Ray Y. W. Lam1, Henry C.B. Chan1, Member, IEEE, Victor O. K. Li2, Fellow, IEEE, Tharam S. Dillon3, Fellow, IEEE, and Victor C. M. Leung4, Senior Member, IEEE

1Department of Computing, Hong Kong Polytechnic University, Hong Kong
2Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong
3Department of Computer Science and Computer Engineering, La Trobe University, Australia
4Department 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 analyzes 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 (FE), 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...
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 1st cycle and the 5th 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 time $t$ and $t+1$ as the $t$th routing cycle. Consider that a sender sets up a communication session of duration $d$ 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 paths. 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 costs.

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

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

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

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

$$\text{Cost}(c,s,a) = \begin{cases} c, & \text{if } a = 0 \\ c - s + C_{\text{switch}}, & \text{if } a = 1 \end{cases}$$

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

$$P((k,j),(i,j),a) = \begin{cases} \sum_{c \in N(i)} N(c), & \text{if } k = l, l = 0 \text{ and } a = 0 \\ \sum_{c \in N(k)} P((i,j),(k,l),a), & \text{if } k = l, l = 0 \text{ and } a = 1 \\ N(k-l), & \text{if } k = l, l > 0 \text{ and } a = 0 \\ 0, & \text{otherwise} \end{cases}$$

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 $N(c)$ for all $c_i \geq k$. The second term represents the cases with cost savings.

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

$$\nu(c,s) = \min_{a \in A(c,s)} \left\{ \text{Cost}(c,s,a) + \sum_{(i,j) \in S_t} P((i,j),(c,s),a) \nu^*(i,j) \right\}$$

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

$$\nu^*(c,x) = 0.$$

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

At each decision epoch $t$, the sender's active packet needs to select an action $a \in A(c,s)$ to make $\nu(c,s)$ as small as
possible. Let \( A^{r}_{(c,s)} \) be the preferred action for state \((c, s)\) at \( t \). This means that it is the action with the smallest \( v'(c, s) \). By using the backward induction algorithm, we can obtain the minimal expected cost for the whole session (i.e., \( v'(c, s) \)) for each initial state \((c, s)\) \( \in S_t \). 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, each initial state \((c, s)\) \( \in S_t \). When the network state is \( (3, 1) \) and the chosen action is to "Switch", the probability that the next state is \((2, 0)\) is \( N(2) + N(3) = 2/3 \). Assuming that \( C_{\text{switch}} = 2 \), the corresponding value of \( \text{Cost}(3, 1, 1) \) is \( 3 - 1 + 2 = 4 \). Other transitional probabilities and \( \text{Cost}(c, s, a) \) can be calculated similarly.

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

\[
v'(c, s, a) = \text{Cost}(c, s, a) + \sum_{j}(P(i, j) E(c, s) a v''(i, j)) (7)
\]

Here, we demonstrate how backward induction is carried out at each decision epoch \( t \).

Step 1. Initialize \( t = 3 \) and \( v'(c, s) = 0 \), for all \((c, s) \in S_h \).

Step 2. For \( t = 2 \), compute

\[
v'(2, 1, 0) = \text{Cost}(2, 1, 0) + \sum_{j} P((2, 0), (2, 1, 0), (2, 0)) v''(2, 0)
\]

\[
v'(2, 1, 1) = \text{Cost}(2, 1, 1) + \sum_{j} P((0, 1), (2, 0), (2, 1, 1)) v''(2, 0) = 3
\]

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

**Table 1** Summary at the First (in brackets) and Second Decision Epochs

<table>
<thead>
<tr>
<th>((c, s))</th>
<th>((1, 0))</th>
<th>((2, 0))</th>
<th>((2, 1))</th>
<th>((3, 0))</th>
<th>((3, 1))</th>
<th>((3, 2))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(v'(c, s))</td>
<td>0</td>
<td>(1)</td>
<td>(2)</td>
<td>(4)</td>
<td>(2)</td>
<td>(4)</td>
</tr>
<tr>
<td>(v'(c, s, a))</td>
<td>0</td>
<td>(1)</td>
<td>(2)</td>
<td>(4)</td>
<td>(2)</td>
<td>(4)</td>
</tr>
<tr>
<td>(A(c, s, a))</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Step 3. Finally compute \( v'(c, s, a) \) and \( A_{(c,s,a)} \) for each state \((c, s) \in S_{t-1} \) as given by the terms in brackets in Table 1.

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

\[
c_j = \mu - \frac{(n-1)\sigma^2}{2} + j - 1 \sigma \quad j = 1, 2, ..., n.
\]

\( N(c) \) is given by the area under the respective normal curve, which is bounded by \( x_j \pm \Delta \) where \( \Delta \) equals a half interval (i.e., \( \delta / 2 \)). 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

\[
C = \frac{d_\mu}{C_{\text{switch}}}. \quad (9)
\]

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

\[
C = \mu + C_{\text{switch}}. \quad (10)
\]

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

\[
C = \sum_{j} v'(c_j) d_{(c_j)}. \quad (11)
\]

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 \( 2/13 + 4/13 + 6/13 = 4 \). Besides the total cost, it is interesting to study how different parameters could affect the decision policy. Fig. 2a shows the preferred actions \( A_{(c,s,a)} \) for states \((250, 0), (250, 25), \ldots , (250, 50)\) at \( t = 1, 2, \ldots , 25 \), when \( \mu = 250 \), \( \sigma = 10 \), \( d = 25 \) and \( C_{\text{switch}} = 500 \). 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:

\[
v'(c, s) = \begin{cases} \text{Cost}((c, s, t)) + \sum_{i \in A} P(i, j)(c, s) i \mu i^t(i, j) & \text{if } s > 0 \\ \text{Cost}((c, s, 0)) + \sum_{i \in A} P(i, j)(c, s) 0 i \mu i^t(i, j) & \text{if } s = 0 \end{cases}
\]

(12)

for \( t = 1, 2, \ldots, d \).

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

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 \( C_{\text{switch}} \) are both small. As \( \sigma \) increases, the average cost decreases in DCR and ACT. When \( C_{\text{switch}} \) is small, DCR and ACT have almost the same average cost. Again, the average cost increases significantly in DYN and DCR as \( C_{\text{switch}} \) increases. However, the average cost remains at a low value in ACT even when \( C_{\text{switch}} \) is large. Furthermore, DCR has a lower average cost than FIX when \( \sigma \) and \( C_{\text{switch}} \) are both large.

To carry out the analysis and comparison, we set our base parameters to the following: \( \mu = 10,000 \), \( \sigma = 2,000 \), \( d = 200 \) and \( C_{\text{switch}} = 200,000 \). 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 \( C_{\text{switch}} \) are both small. However, as \( C_{\text{switch}} \) 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 \( C_{\text{switch}} \) are large.
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 that a lower network cost can be reserved.

Table II: Summary of the Analysis

<table>
<thead>
<tr>
<th>Increase in ( \mu )</th>
<th>Effect on expected total cost</th>
<th>Effect on threshold</th>
<th>Preferred schemes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIX</td>
<td>DYN</td>
<td>DCR, ACT</td>
<td>DCR, ACT</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>(-)</td>
<td>(-)</td>
<td>DCR, ACT</td>
</tr>
<tr>
<td>( d )</td>
<td>( t_2 )</td>
<td>( t_3 )</td>
<td>DCR, ACT</td>
</tr>
<tr>
<td>( C_{\text{min}} )</td>
<td>( t_1 )</td>
<td>( t_2 )</td>
<td>FIX, ACT</td>
</tr>
</tbody>
</table>

Key:
- \( \uparrow \): increase the value
- \( \downarrow \): decrease the value
- \( \dagger \): no effect on the value

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


2379