

COMPUTING MDP COST FUNCTION FOR HIGH SPEED NETWORKS WITH SAMPLE-PATH AND QUANTIZATION*

XI-REN CAO[†], JUNJIE WANG[†], AND CHIN-TAU LEA[†]

Abstract. The theory of Markov decision process (MDP) is a promising tool in many network-related problems, including routing, admission control, and capacity planning. One problem often encountered is the lack of exact information about the state transition rates. In addition, the application of MDP in a high-speed network environment is hindered by a large state space and consequently a prohibitive computational complexity.

In this paper, we apply two techniques to solve the problem mentioned above. The sample-path-based performance potential theory removes the need of knowing the exact system parameters. Cost quantization, on the other hand, can reduce the state space to a manageable size. By combining the two, we are able to obtain the efficient solution to the cost computation problem in a high-speed network. Numerical results show that considerable computational resources can be saved without degrading the performance significantly.

Keywords: Cost-based routing, Markov decision process, Performance potentials, Quantization.

1. Introduction. The theory of Markov decision process (MDP) [1]-[4] is a promising tool in many network-related problems. For example, cost-based routing has been widely studied in networking research in which the cost function of adding a new connection can be captured by MDP theory [5]-[9]. Cost-based routing attracts research attention not only because of its high throughput, but also because of its ability of removing the unstable conditions inherent in non-hierarchical routing networks [10]. It can be implemented in a centralized or distributed fashion.

In spite of the advantages offered by MDP-based schemes, there are several major hurdles to overcome before MDP-based techniques can be applied to solve high-speed network problems. The hurdles are all related to the basic task in cost-based routing by the MDP approach: cost computation. First, the specific information about the system, such as transition rates and steady state probabilities that are required for computing the cost function, is sometimes missing or varying with time. Second, the conventional cost computation methods have a high complexity because they rest on the computation-based MDP approaches, which involve large recursive equations or the inverse of large-dimensional matrices. Third, the computational complexity is determined by the number of states. In a high-speed network the state space is generally enormous, thereby making the cost computation and the implementation of MDP-based schemes an impossible task. (As an example, a 1 Gbps link can sustain 5×10^5 calls, and thus the same number of states will exist.)

Contrary to the traditional computation-based approach in MDP theory, the

*Received Oct 26, 2000; accepted for publication Mar 7, 2001. This work was supported by the Hong Kong RGC Grant HKUST 6203/98E and Grant CRC98/01.EG15

[†]Department of Electrical and Electronic Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, Email: eecao@ust.hk; jwang@redswitch.com, eelea@ust.hk

performance potential theory [11][12] provides a sample-path-based approach. This methodology received more and more attention recently for its promising application against some of the aforementioned difficulties in the conventional computation-based MDP approaches. The single-sample-path-based potential theory can be an appealing candidate for cost computation in cost-based routing in the sense that it can elegantly alleviate the first two problems mentioned above. The concept of performance potential offers a desirable property that the potential can be measured in a single sample path without any particular knowledge about the system parameters. Therefore, the online estimation can adapt to the dynamic features of the systems such as high-speed networks. Some simple and efficient algorithms have been developed for online implementation [12].

Another development related to cost computation in MDP routing is the concept of cost quantization. In high-speed network related problems, the cost function changes little among adjacent states. In a sense, the cost function is like a continuous function. It was shown in [13] that we can quantize the cost function into a small number of levels without sacrificing the accuracy significantly. The result of quantization is a considerable reduction in the number of states required in cost computation. This technique can solve the third problem mentioned above. But quantization described in [13] was developed based on the conventional cost computation schemes which still requires system information like arrival rates, and steady state probabilities.

In this paper, we combine the advantages of the two, sample-path-based MDP approach and quantization, to remove the major obstacles mentioned above. We will show that this technique will not degrade the estimation accuracy much, but save computational resources significantly. The paper is organized as follows. In Section 2, we briefly review the MDP model and the potential theory. We will explore the notion of performance potential in order to acquire the basis of understanding the cost quantization. In Section 3, we discuss the cost quantization by the method of state aggregation. We will focus on the sample-path-based approach and develop some algorithms to estimate quantized costs. Some numerical results are shown in Section 4. It is illustrated that a significant amount of computational resource can be saved by state aggregation without effecting the estimation accuracy. Finally, we conclude the paper in Section 5.

2. Markov Decision Process and Potential Theory. In this section, we will briefly review the performance potential theory in MDP model. We assume that the Markov chain, denoted as $\mathbf{X} = \{X_n, n \geq 0\}$, has a finite state space $\mathcal{S} = \{1, 2, \dots, M\}$. At any state $i \in \mathcal{S}$, an action α is taken from an action space $\mathcal{A}(i)$ and is applied to the Markov chain. We assume that the number of actions is finite, and we only consider stationary policies. A stationary policy is a mapping $\mathcal{L} : \mathcal{S} \rightarrow \mathcal{A}$, which determines the underlying transition probability matrix $P^{\mathcal{L}} = [P^{\mathcal{L}(i)}(i, j)]_{i=1}^M |_{j=1}^M$. Let ε be the policy space, for simplicity, we assume the Markov chain under any policy $\mathcal{L} \in \varepsilon$ is ergodic.

Accordingly, the steady state probability corresponding to policy \mathcal{L} is denoted as a vector $\pi^{\mathcal{L}} = [\pi^{\mathcal{L}}(1), \dots, \pi^{\mathcal{L}}(M)]$. Suppose that at each stage with state i and

control action $\alpha \in \mathcal{A}(i)$, a reward $f(i, \alpha)$ is received, and we define vector $f^{\mathcal{L}} = [f(1, \mathcal{L}(1)), \dots, f(M, \mathcal{L}(M))]^T$. The long term expected reward corresponding to policy \mathcal{L} is then

$$(1) \quad \eta^{\mathcal{L}} = \lim_{N \rightarrow \infty} \frac{1}{N} E \left\{ \sum_{n=0}^{N-1} f[X_n, \mathcal{L}(X_n)] \right\},$$

where “ E ” denotes the expectation. For ergodic chains, the above limit exists and does not depend on the initial state.

It is shown in [11] that, the concept of performance potentials leads directly to the solution of MDP. Most results in infinite-horizon, average-cost MDP can be derived from this notion. Formally, the performance potential vector g is defined by Poisson equation (the superscript indicating the policy is dropped for simplicity)

$$(2) \quad (I - P + e\pi)g = f$$

where I is the identity matrix and the M -dimensional vector $e = (1, \dots, 1)^T$.

The potential can be derived by solving the above equation, which is usually referred to as computation-based approaches. Unfortunately, we notice that a matrix inverse is required, which may be highly computationally complex, especially when the cardinality of the state space \mathcal{S} is large. Moreover, in real systems, some elements in the transition probability matrix P may be unknown, thereby making the computation even impossible. For example, in the call admission and routing problems, the transition probabilities depend on the arrival rate and the holding time of the call, which may be unknown or time-varying.

The most appealing characteristic of the performance potential lies in its desirable physical property. That is, performance potentials can be measured in a single sample path during the system evolution. It is proved in [11]

$$(3) \quad g(i) = \lim_{N \rightarrow \infty} E \left\{ \sum_{n=0}^{N-1} [f(X_n) - \eta] | X_0 = i \right\}.$$

Therefore, potentials can be measured according to (3) along a sample path.

Furthermore, let

$$(4) \quad d(i, j) = g(j) - g(i), \quad i, j = 1, 2, \dots, M,$$

and $D = eg^T - g e^T$. In [11], $d(i, j)$ is called *realization factor* and D is called *realization matrix*. Let \mathbf{X} and \mathbf{X}' be two independent Markov chains with the same transition probability matrix and the reward function, but different initial states $X_0 = i, X'_0 = j$. We define $N_{ij} = \min\{n : n \geq 0, X_n = X'_n\}$, i.e. at stage N_{ij} the two chains merge for the first time. We have [11]

$$d(i, j) = E \left\{ \sum_{n=0}^{N_{ij}} f(X'_n) - f(X_n) | X_0 = i, X'_0 = j \right\}$$

This equation provides a physical interpretation of the relative factors. That is, $d(i, j)$ measures the effect of the change from state i to j on the system performance.

Now consider a Markov chain $\mathbf{X} = \{X_n, n \geq 0\}$ starting with $X_0 = j$. Let $L_j(i) = \min\{n : n \geq 0, X_n = i\}$, when the Markov chain reaches state i for the first time. It is proved in [11] that

$$(5) \quad d(i, j) = E\left\{ \sum_{n=0}^{L_j(i)-1} [f(X_n) - \eta] \mid X_0 = j \right\}.$$

Eqn. (5) offers another explanation of $d(i, j)$ based on a single-sample-path-based observation. It also suggests another method to estimate the performance potential in addition to (3). The estimation based on (3) or (5) requires no knowledge about the system structure (such as transition probability matrix) as long as the sample path is observable. This characteristic essentially converts the computation-based methodology to a sample-path-based approach. Some simple and efficient algorithms for estimating potentials based on Eqn. (3) and (5) were developed in [12]. In this paper, $d(i, j)$ is also called *relative cost* or simply *cost*, which represents the consequence imposed on the system performance because of the transition from state j to i .

The performance potential plays an important role in optimizing the infinite-horizon, average-cost MDP. After it is derived by computation or online estimation, it can be applied in policy making to achieve the optimal policy. Some algorithms are developed to perform the optimization in iterative procedures [12]. It is also guaranteed that the iteration will converge to the optimal policy with probability one. Interested readers can refer to [11] and [12] for details. In this paper, however, we don't consider the optimization. Instead, we focus on the efficient derivation of the potentials in large state space, which can be straightforwardly applied to the optimization of cost-based routing problems.

3. Cost Quantization. As in almost all routing problems in a large network, the link-independent assumption is always used to tame the complexity of the problem. With this assumption, we can focus on a single link in the routing network to investigate the cost computation problem.

Let us assume that the arrival of the call requests is governed by Poisson process with rate λ and the holding time of a call is exponentially distributed with the mean $1/\mu$ seconds. Take the number of the existing calls on the link as the state variable, the behavior of the link can be well formulated as a Markov chain \mathbf{X} with a state space $\mathcal{S} = \{1, 2, \dots, M\}$.

In network problems, like routing and admission control, we are interested in the specific cost function defined as

$$(6) \quad C(i) = d(i+1, i) = g(i) - g(i+1),$$

for $i = 1, \dots, M-1$. Hence, $C(i)$ measures the effect of accepting a call at state i on the long-term system reward. The combined cost of all links along a particular path will

be compared with the reward of establishing a call to determine the action. Clearly, the problem of cost computation is equivalent to the computation of potentials.

The sample-path-based technique described in section 2 allows us to obtain $g(i)$ using online estimation, thereby removing the requirement of knowing the transition rates and steady-state probabilities. But the enormous number of states make the approach impossible to implement in a high-speed network environment. For example, a 1 Gbps link can have as many as 0.5 million states for supporting voice channels. An intractably long sample path is required to guarantee that all the states are sufficiently visited for an acceptable estimation accuracy. This, however, conflicts with the requirement of real-time control in high-speed networks.

Quantization is a concept proposed to tackle this problem [13]. It can be performed at two levels. One is called *rate quantization*, where the continuous bandwidth requirement of various calls is discretized into several levels. A customer of a quantized-rate network will be forced to subscribe to the next higher rate if his required rate is not supported. The result is a considerable reduction in the number of *dimensions*. Past studies have shown that rate quantization can be done with only a handful of rates without significantly degrading the throughput of the network.

The second level is *cost quantization* which leads to a significant reduction in the number of states in each dimension by state aggregation. In this paper, we consider the case where the network supports only one rate (thus one dimension in the Markov model), thereby focusing on the level of cost quantization. We will demonstrate that cost quantization can be combined with sample-path techniques to reduce the computational complexity. It is illustrated in [10] that multi-rate problems can be solved in a similar manner.

3.1. State Aggregation. Fig. 3.1 shows a graph of performance potentials under a particular policy. A close look reveals that the potential has a wide flat region and can be closely approximated with a quantized function with m levels.

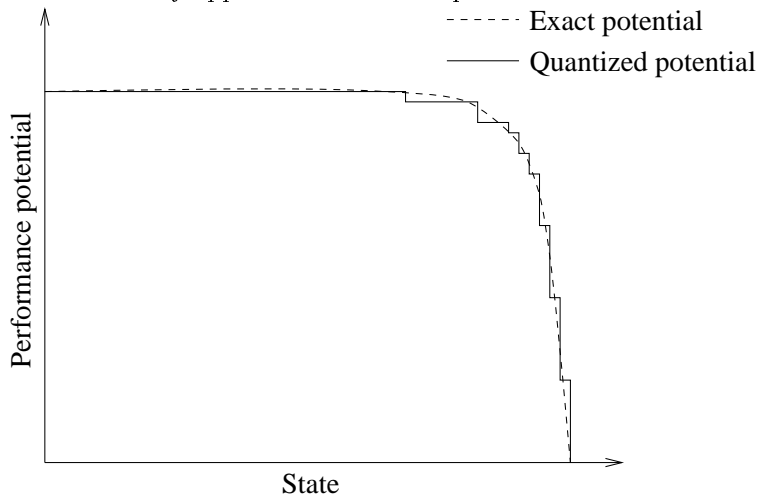


FIG. 3.1. *Performance potentials*

Quantizing potentials into m levels is equivalent to assuming the potentials of all states i , where $j_k < i \leq j_{k+1}$, are the same and equal to that of the state j_{k+1} . These states are then aggregated into a single state as shown in Fig. 3.2.

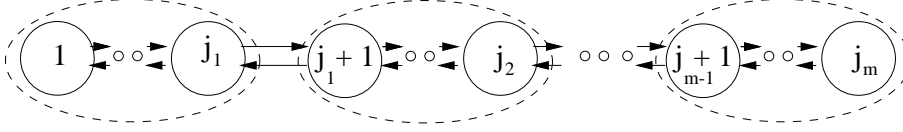


FIG. 3.2. State aggregation

Considering the quantized potentials (as plotted in solid line in Fig. 3.1), we can construct a new Markov model by grouping all states with the same potential into one without changing the long term reward. After that, the original cost function can be reconstructed with linear interpolation done for the quantized cost function. The original theory of cost quantization was developed for a conventional computation-based MDP approach [13]. In the following we will show it can be applied in the same way to the sample-path based approach.

Let's suppose that the performance potentials are the same in a certain range, that is, $g(i) = g(j)$, where $i, j \in K$ and $K = \{1, 2, \dots, k\}$. Therefore, we can aggregate state $1, \dots, k$ into one state called 0, then we construct an aggregated Markov chain \mathbf{Y} with a state space $\mathcal{S}_Y = \{0, k+1, \dots, M\}$ of cardinality of $(M - k + 1)$.¹ Here, we refer to k as the *aggregation ratio* to indicate the extent of the state compression. Accordingly, we have the aggregated potential vector $g_Y = [g_Y(0), g_Y(k+1), \dots, g_Y(M)]$ and the steady state probabilities $\pi_Y = [\pi_Y(0), \pi_Y(k+1), \dots, \pi_Y(M)]$. By the conservation law of transition rate, the transition probability matrix P_Y is defined as:

$$(7) \quad P_Y(j, 0) = \sum_{m=1}^k P_X(j, m)$$

$$(8) \quad P_Y(0, j) = \frac{\sum_{m=1}^k P_X(m, j) \pi_X(m)}{\sum_{m=1}^k \pi_X(m)}$$

$$(9) \quad P_Y(0, 0) = \frac{\sum_{m=1}^k \sum_{n=1}^k P_X(m, n) \pi_X(m)}{\sum_{m=1}^k \pi_X(m)}$$

for $j = k+1, \dots, M$, and

$$(10) \quad P_Y(i, j) = P_X(i, j),$$

for $i, j = k+1, \dots, M$.

The reward function of chain \mathbf{Y} is defined as:

$$(11) \quad f_Y(i) = \begin{cases} \frac{\sum_{m=1}^k \pi_X(m) f_X(m)}{\sum_{m=1}^k \pi_X(m)}, & \text{when } i = 0 \\ f_X(i), & \text{when } i = k+1, \dots, M \end{cases}$$

¹To avoid confusion, we use the subscript \mathbf{X} and \mathbf{Y} to indicate the variables based on the definition of the original and aggregated Markov chain, respectively.

It is easy to verify from Eqn. (11) that the long-term reward $\eta_Y = \eta_X$.

Next to the above formulation, we have the following theorem.

Theorem: Assume that the potential of the Markov chain \mathbf{X} satisfies: $g_X(i) = g_X(j)$, for $i, j = 1, \dots, k$. We construct a new Markov chain \mathbf{Y} by aggregating state $1, 2, \dots, k$ into one state denoted as 0 according to (7)-(11), then we have:

$$(12) \quad g_X(i) = \begin{cases} g_Y(0), & \text{when } i = 1, 2, \dots, k \\ g_Y(i), & \text{when } i = k + 1, \dots, M. \end{cases}$$

Proof: To simplify the notation, we let $g_X(i) = \bar{g}$, for $i = 1, \dots, k$, and $\eta_X = \eta_Y = \eta$.

From the Poisson equation

$$(13) \quad (I - P_X + e \pi_X) g_X = f_X.$$

We expand the first k equations as

$$(14) \quad \bar{g} - \bar{g} \sum_{j=1}^k P_X(i, j) - \sum_{j=k+1}^M P_X(i, j) g_X(j) + \eta = f_X(i),$$

for $i = 1, \dots, k$.

Multiplying the i -th equation by $\pi_X(i)$ and add them up, we have

$$(15) \quad \begin{aligned} \bar{g} \sum_{i=1}^k \pi_X(i) - \bar{g} \sum_{i=1}^k \sum_{j=1}^k P_X(i, j) \pi_X(i) - \sum_{j=k+1}^M \sum_{i=1}^k [P_X(i, j) \pi_X(i)] g_X(j) \\ + \eta \sum_{i=1}^k \pi_X(i) = \sum_{i=1}^k \pi_X(i) f_X(i). \end{aligned}$$

Dividing (15) by $\sum_{i=1}^k \pi(i)$ and applying Eqn (7)- (11), we have

$$(16) \quad \bar{g} - P_Y(0, 0) \bar{g} - \sum_{j=k+1}^M P_Y(0, j) g_X(j) + \eta = f_Y(0).$$

Considering the other $(M - k)$ equations in (13),

$$(17) \quad g_X(i) - \bar{g} \sum_{j=1}^k P_X(i, j) - \sum_{j=k+1}^M P_X(i, j) g_X(j) + \eta = f_X(i),$$

for $i = k + 1, \dots, M$.

Applying (9) and (10), (17) can be written as

$$(18) \quad g_X(i) - P_Y(i, 0) \bar{g} - \sum_{j=k+1}^M P_Y(i, j) g_X(j) + \eta = f_Y(i).$$

Putting (16) and (18) in matrix form, we have

$$(19) \quad (I - P_Y + e \pi_Y) g_Y = f_Y.$$

where $g_Y = [\bar{g}, g_X(k+1), \dots, g_X(M)]^T$. Hence we have (12).

Following this theorem, we can convert the quantized potentials g_Y to the original g_X . After that, we can compute the cost function according to (6). The potential of the aggregated chain can be obtained by online estimation using the algorithms developed in the next section.

Note that in the theorem, we assume that $g_X(i) = g_X(j)$ in a range of $i, j \in K$. However, this may be not absolutely true in practice. Therefore, the state aggregation based on the theorem will introduce some error, which is called *quantization loss* in [13]. The point is, the quantization loss under a proper grouping scheme is acceptably small compared with the computational savings [13]. We will evaluate the performance of quantization schemes in terms of the quantization error.

3.2. Aggregated Potential Estimation. In computation-based approach, we can simply obtain the potential g_Y by solving Poisson equation (19). In this paper, however, we will mainly discuss the sample-path-based approach for potential estimation. Furthermore, we will demonstrate that the potential for aggregated chain \mathbf{Y} can be measured directly by observing the sample path of \mathbf{X} . With the state aggregation, we will significantly reduce the number of states and make the storage and database management tractable.

Considering Eqn.(3) and (12), it clearly follows that

$$(20) \quad g_Y(i) = \begin{cases} \lim_{N \rightarrow \infty} E[\sum_{n=0}^{N-1} f_X(X_n) - \eta | X_0 \in K], & \text{when } i = 0 \\ \lim_{N \rightarrow \infty} E[\sum_{n=0}^{N-1} f_X(X_n) - \eta | X_0 = i], & \text{when } i = k+1, \dots, M. \end{cases}$$

This equation offers a similar method to (3) to online estimate g_Y . From (20), we choose an integer L and define

$$(21) \quad g_Y^L(i) = E[\sum_{n=0}^{L-1} f_X(X_n) | X_0 = i] - L\eta$$

for simplicity, we force $X_0 = 0$, if $X_0 \in K$. Hence we have

$$(22) \quad g_Y^L(i) = \lim_{N \rightarrow \infty} \left\{ \frac{\sum_{n=0}^{N-L} I_i(X_n) [\sum_{m=0}^{L-1} f_X(X_{n+m})]}{\sum_{n=0}^{N-L} I_i(X_n)} - \frac{L}{N} \sum_{n=0}^{N-1} f_X(X_n) \right\}, \quad w.p.1$$

for $i = 0, k+1, \dots, M$. $I_i(\cdot)$ is the indicator function, that is, $I_i(X_n) = 1$, if $X_n = i$; or $I_i(X_n) = 0$, otherwise. Consequently, we have the following algorithm.

Algorithm 1

1. Choose two fixed integers L and N .
2. Estimate $g_Y^{L,N}(i)$ by

$$(23) \quad g_Y^{L,N}(i) = \frac{\sum_{n=0}^{N-L} I_i(X_n) [\sum_{m=0}^{L-1} f_X(X_{n+m})]}{\sum_{n=0}^{N-L} I_i(X_n)} - \frac{L}{N} \sum_{n=0}^{N-1} f_X(X_n),$$

for $i = 0, k+1, \dots, M$.

From (3) and (22), it follows

$$(24) \quad \lim_{L \rightarrow \infty} \lim_{N \rightarrow \infty} g_Y^{L,N}(i) = g_Y(i), \quad w.p.1.$$

Furthermore, we define the realization factor in \mathbf{Y}

$$(25) \quad \begin{aligned} d_Y(i, 0) &= g_Y(0) - g_Y(i) \\ &= g_X(j) - g_X(i) \\ &= d_X(i, j), \end{aligned}$$

for $i = k + 1, \dots, M$ and $j \in K$.

Similarly to the estimation based on (5), we observe a sample path starting from $X_0 = i$. Let $\mu_0 = 0$ and $\mu_{k+1} = \min\{n : n > \mu_k, X_n = i\}$, $k \geq 0$. Then $\mu_k, k \geq 0$ are regenerative points. We define $v_k(0) = \min\{n : \mu_{k+1} > n > \mu_k, X_n \in K\}$.

Based on Eqn. (5) and (25), we have

$$(26) \quad d_Y(i, 0) = \lim_{k \rightarrow \infty} \frac{1}{\sum_{k=0}^{K-1} \chi_k(0)} \left\{ \left[\sum_{k=0}^{K-1} \chi_k(0) \sum_{n=v_k(0)}^{\mu_{k+1}-1} f_X(X_n) \right] - \left[\sum_{k=0}^{K-1} \chi_k(0) [\mu_{k+1} - v_k(0)] \right] \eta \right\},$$

for $i = k + 1, \dots, M$, where

$$\chi_k(0) = \begin{cases} 1, & \text{if } \{\mu_{k+1} > n > \mu_k, X_n \in K\} \neq \Phi \\ 0, & \text{otherwise.} \end{cases}$$

Therefore, we have the following algorithm. In the algorithm, we assume that the Markov chain starts from $X_0 = i$. (if $i \in K$, we denote $X_0 = 0$.)

Algorithm 2

1. Create six $(M-k+1)$ -dimension vectors a, b, c, d, h, l and w , set $a := 0, b := 0, c := 0, h := 0, l := 0$, and $w := 0$, denote their j th components as $a(j), b(j), c(j), d(j), h(j), l(j)$ and $w(j)$, $= j = 0, k + 1, \dots, M$, respectively. Set a scale variable $u := 0$.
2. At the n th transition of the Markov chain, do:
 - (a) Set $u := u + f_X(X_n)$;
 - (b) If $X_n \in K$, set $Y := 0$; otherwise, $Y := X_n$.
 - (c) If $Y \neq i$, then
 - i. if $a(Y) = 0$, set $a(Y) := 1$;
 - ii. for $j = 0, k + 1, \dots, M$, if $a(j) = 1$, then $b(j) := b(j) + f_X(X_n)$, $h(j) := h(j) + 1$;
 - (d) If $Y = i$, then for all $j \neq i$,
 - i. set $c(j) := c(j) + b(j)$ and $l(j) := l(j) + h(j)$;
 - ii. if $h(j) > 0$, set $w(j) := w(j) + 1$;
 - iii. for $j = 0, k + 1, \dots, M$, set $a(j) := 0, b(j) := 0$, and $h(j) := 0$;

3. At the last stage N , set $\eta = \frac{u}{N}$ and $g(i) = 0$, for all $j = 0, k + 1, \dots, M$, and $j \neq i$,

$$g_Y(j) = \frac{c(j) - l(j)\eta}{w(j)}$$

From this algorithm, we can see that the estimation based on the aggregated state space need only six $(M - k + 1)$ -dimensioned vectors instead of M -dimensioned vectors. It can be easily extended to the case where we further aggregate the states to m levels. Clearly, the less the m , the less is the storage required. In addition to the memory savings, state aggregation is also expected to increase the steadiness of the estimation against the length of the sample path. Generally, the estimation errors result from the insufficient length of the sample path. However, by state aggregation, hitting any element of an aggregated state will contribute to the estimation, thereby enhancing the accuracy of the estimation, especially under short sample paths. This intuition is verified by the numerical results in Section 4.

Note that in the practical state aggregation, the states within a group are not necessary to have exactly the same potentials. Therefore, the ideal conversion from g_Y to g_X according to (12), which is based on the assumption of perfectly flat region of potential values, cannot be performed. In order to recover the original potential from the quantized value, we apply the linear interpolation as shown in Fig. 3.3.

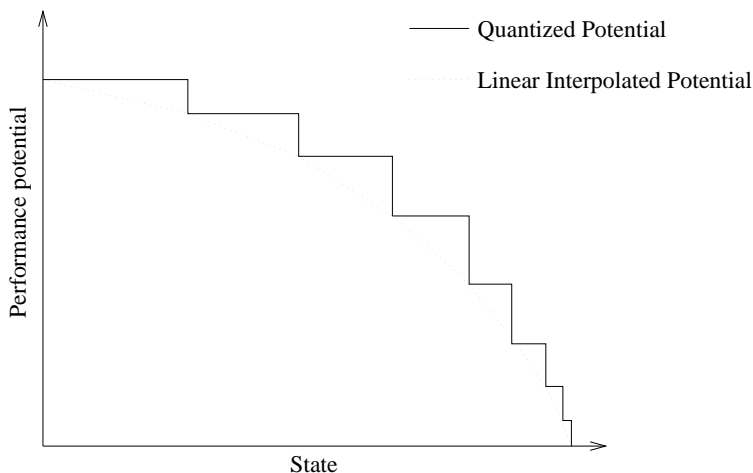


FIG. 3.3. *Linear interpolation*

Accordingly, the interpolated potential can be yielded as

$$(27) \quad g_X(i) = \frac{[g_Y(j_{k-1}) j_k - g_Y(j_k) j_{k-1}] + i [g_Y(j_k) - g_Y(j_{k-1})]}{j_k - j_{k-1}}.$$

4. Examples. The example presented below is a common routing problem in a connection-oriented network such as ATM. One class of routing algorithms is cost-based. In such a routing algorithm, whether a connection is rejected or not is based on the sum of the "cost" of all the links of the entire path. When the cost exceeds

its "reward" of the connection, the connection will be rejected. The cost and reward of a connection in a connection-oriented network, like ATM, are normally defined and computed from the concept of MDP. Cost-based routing has many advantages—like high efficiency and high stability. But the main problem is two fold: (1) the complexity of the cost computation is proportional to the number of states, and (2) the traffic pattern is not totally stationary. The technique proposed in this paper is intended to solve the two problems. Quantization can solve the first problem and the sampling path (measurement) technique solves the non-stationary problem.

We give an example below to illustrate our technique. A network consists of many nodes and links (Fig. 4.1). Although all links and nodes are dependent, a necessary technique to make the problem tractable is to assume all links and nodes are independent. Since a connection's bandwidth is usually much smaller than the total bandwidth of a link, this assumption has been shown to be a good approximation [7]. With this link-independent assumption we can derive the total amount of traffic generated in each node from the routing policy we use [7] and the cost computation in network routing is reduced to cost computation for one link. Thus the example below, although for one link, is applied to the whole network routing problem.

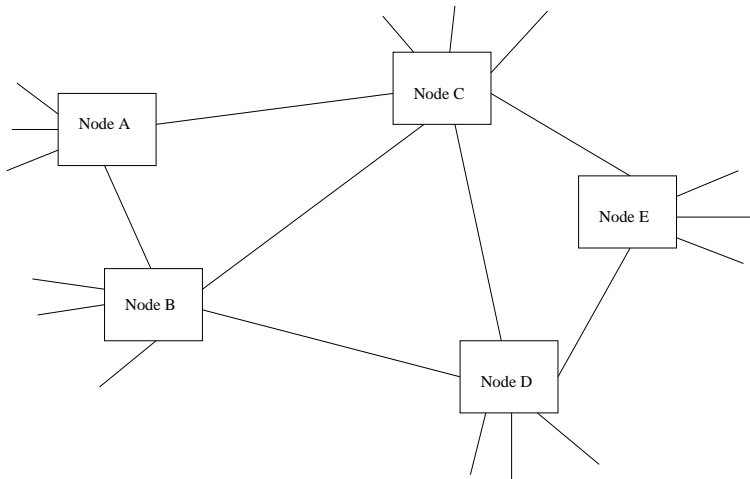


FIG. 4.1. *Link and nodes are normally assumed to be independent in network routing.*

Consider the example shown in Fig. 4.2. The link has a capacity of 1000 units and each call will take one unit. Call arrivals are assumed to be Poisson, but the rate is not known and will be measured on line. Although the discussion is only for one link, keep in mind the cost computation in a network is similar since link-independent assumption is normally used; otherwise, the problem becomes intractable.

4.1. How to Measure the Accuracy of the Algorithm. To measure the error in online estimation, we evaluate the ratio of mean squared cost (MSC) to

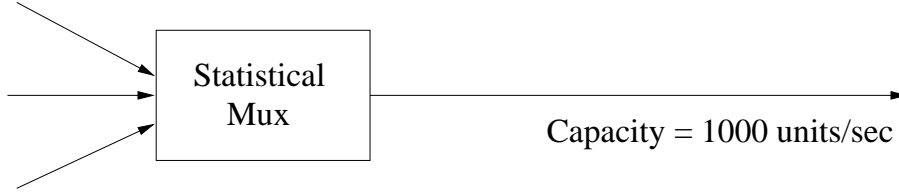


FIG. 4.2. Example of a statistical multiplexer

mean squared error (MSE) as the performance metric. That is,

$$MSC = \sum_{i=1}^M \pi_X(i) C^2(i)$$

$$MSE = \sum_{i=1}^M \pi_X(i) [C(i) - \bar{C}(i)]^2$$

where $C(i)$ and $\bar{C}(i)$ are the theoretical and the estimated cost functions, respectively.

Similar to the *signal-to-noise ratio* (SNR) used in signal detection and estimation, we define the *cost-to-quantization error ratio* ($CQER$)

$$CQER = \frac{MSC}{MSE}$$

as the performance measure to investigate the quantization error. The $CQER$ will be represented in dB (i.e. $10 \times \log_{10} CQER$).

We need to point out that, the error measured by $CQER$ results from two different sources: one is the quantization loss, the other is the estimation error attributed to the insufficient sample path². We separate these two kinds of errors in our study to examine the effectiveness of (i). single-sample -path based technique, and (ii). cost quantization under particular aggregation schemes.

4.2. Aggregation Strategy. In [13], an aggregation strategy named *Backward Uniform Quantization* (BUQ) was proposed for its simplicity. In this paper, we develop another grouping strategy called *Difference Dependent Quantization* (DDQ). This scheme is motivated by the observation that the difference of potentials is monotonically decreasing (as can be seen from Fig. 3.1). The basic idea of DDQ is to select a high aggregation ratio at the flat region and a low aggregation ratio at the steep region. Intuitively, this scheme will cause less quantization loss and it is verified by our simulation results. In the simulation, we assume the link capacity is 1000 units/sec and the bandwidth requirement of a single call is uniformly 1 unit/sec. Hence the original state space holds 1001 states. We adopted two DDQ methodologies to compress the state space to 39 states and 60 states, respectively. By applying Algorithm 2, the memory required in potential estimation is reduced by 96% and 94%, respectively. Fig. 4.3 contains a sample result of the DDQ scheme.

²In a computation-based cost quantization, we don't have the this source of error.

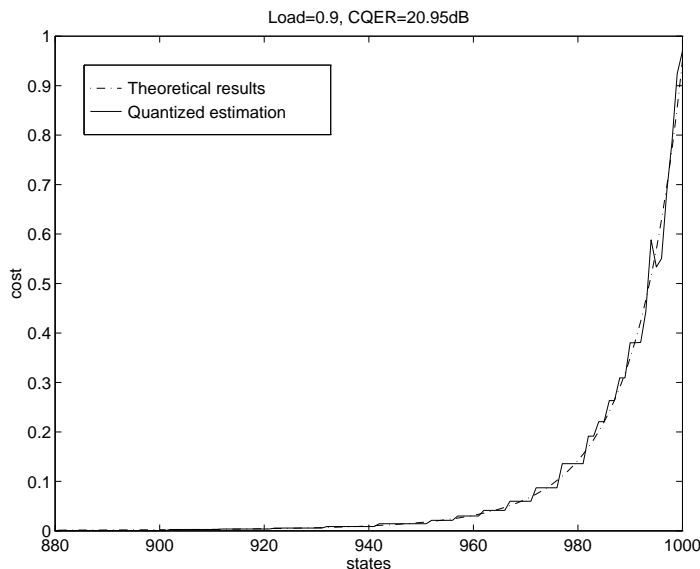


FIG. 4.3. Quantized cost function (quantization level = 39)

We found that DDQ scheme yields a smaller $CQER$ than the BUQ procedure. Some sample results are shown in Table 1. For comparison, we also list the performance of estimation with no quantization. Note that the length of the sample path is measured by the number of arriving calls.

TABLE 4.1
Performance of estimating quantized costs (DDQ scheme)

Load	Length of sample path ($\times 10^6$)	CQER (dB)		
		$m = 39$	$m = 60$	$m = 1001$ (no quantization)
0.9	150	20.95	22.31	21.42
0.9	50	15.4	15.92	15.86
0.9	10	8.7	8.16	8.02
1.0	150	9.95	15.96	31.32
1.0	50	9.95	15.96	26.33
1.0	10	9.93	15.54	21.28
1.0	2.5	9.85	14.63	17.06
1.0	0.5	10.03	11.14	10.29

From Table 1, we can investigate the effects of two factors on the $CQER$. The first is the performance of online estimation approach, which can be found in the last column. Here we remove the effect of quantization in order to separate two kinds of errors mentioned in Section 4.1, i.e. we select quantization level $m = 1001$. It can be seen that online estimation produces accurate results. For example, under the load of 1.0, a sample path of the length of only 0.5×10^6 yields a $CQER$ of 10.29.

The second is the effect of cost quantization on the $CQER$. It can be found that under the load of 0.9, the cost quantization yields nearly the same performance under two different quantization levels (i.e. $m = 39$ and $m = 60$). Moreover, even with no quantization ($m = 1001$), the estimation performance doesn't increase considerably. In other words, a moderate quantization can yield a similar performance as in the non-quantization case, while a considerable amount of memory can be saved. The reason is that, under the load of 0.9, the flat region of the cost curve is relatively wide, therefore the high quantization level doesn't necessarily reduce the quantization loss. In contrast, we will show that this observation does not hold true under the load of 1.0, where the flat region is relatively short. Hence, the higher rate of compression will cause more quantization loss.

Under the load of 1.0, we can find that the quantized cost estimation is more robust than the non-quantized estimation. That is, the $CQER$ degrades quite slowly with the truncation of the sample path while it degrades very fast under no quantization. This is intuitively true because in the aggregated state, visiting to any of its elements will contribute to the estimation, thus making it less sensitive to the length of the sample path. For example, if a $CQER$ of 10 dB is acceptable, the performance of a sample path as long as 5×10^5 can even outweigh the non-quantized estimation.

To choose the length of sample path is a key issue in the estimation. Naturally, a longer sample path will yield more accurate results. However, it is not realistic in practical systems, especially in the dynamic environments with parameters changing relatively fast. The basic approach is to select the length of sample path such that the traffic characteristic in a single sample path remains consistent. Take an example in the telephone network, if the time scale can be approximately divided into several segments (say every one hour) such that the traffic load within each interval can be assumed constant. Thus we can set the sample path (measured in time scale) to be as long as one hour.

5. Conclusion. MDP theory has a wide range of applications to networking problems. The sample-path-based performance potential theory allows us to online compute the cost function in the cost-based routing. It relaxes the requirement of system identification, therefore it is very attractive from a practical point of view in the dynamic systems.

The main obstacle to overcome in a high-speed network environment is the enormous number of states. We overcome this difficulty by applying the concept of quantization by state aggregation. We developed algorithms and a theorem in the paper to establish the foundation for combining quantization and potential theory. Numerical results showed that this technique yields satisfactory accuracy and saves great computational resources as well.

REFERENCES

- [1] R. A. HOWARD, *Dynamic Programming and Markov Processes* John Wiley & Sons, 1960.

- [2] S. STIDHAM AND R. WEBER, *A Survey of Markov Decision Models for Control of Networks of Queues*, *Queueing System*, 13(1993), pp. 291–314.
- [3] M. L. PUTERMAN, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Willy, New York, 1994.
- [4] H. C. TIJMS, *Stochastic Modeling and Analysis: A Computational Approach*, Willy, New York, 1986.
- [5] *Special Issue on Dynamic Routing*, *IEEE Communications Magazine*, July 1995.
- [6] C. HUITEMA, *Routing in the Internet*, Prentice Hall, 1995
- [7] A. GIRARD, *Routing and Dimensioning in Circuit-Switched Networks*, Addison-Wesley, 1990.
- [8] K. R. KRISHNAN AND T. J. OTT, *State dependent routing for telephone traffic: theory and results*, Proc. 25th IEEE Decision and Control Conf., Athens, Greece, pp. 2124-2127, Dec. 1986.
- [9] R.-H. HWANG, J. F. KUROSE, AND D. TOWSLEY, *State dependent routing for multirate loss network*, Proc. Globecom'92, pp. 565-570. 1992.
- [10] C. -T. LEA AND K. -W. KE, *On cost-based routing in connection-oriented high-speed networks*, *Journal of High-Speed Networks*, to appear, 2000.
- [11] X. R. CAO AND H. F. CHEN, *Potentials, Perturbation Realization, and Sensitivity Analysis of Markov Processes*, *IEEE Transactions on Automatic Control*, 42(1997), pp. 1382–1393.
- [12] X. R. CAO, *Single Sample Path Based Optimization of Markov Systems*, *Journal of Optimization: Theory and Applications*, 100:3(1999), pp. 527–548.
- [13] C. -T. LEA AND K. -W. KE, *Quantization and Cost Computation in MDP-based Admission and Routing*, Proc. Infocom'96, San Francisco, CA, March 1986.

