NETWORK MODEL CONTROL OF TRAFFIC UPDATES IN DECENTRALIZED ROUTING STRATEGIES TRANSFER AND ITS OPTIMIZATION

Victoria BOGHANASTIUC¹, Veaceslav RÎCU², Alexei VASCAN³, Olesea BERDAGA⁴, Svetlana MUTU⁵

Technical University of Moldova, Automatica Departament,
Bd. Ștefan cel Mare nr.168, MD 2004, Chișinău, Republic of Moldova,
phone: (+373-22) 497694, fax: (+373-22) 497004,
slaviu@mail.utm.md

Abstract. The real time control of traffic routing in asynchronous decentralized communication networks is investigated. This a decentralized routing strategy is obtained which is proved to be stable under suitable assumptions about the external inputs. The decentralized strategy is also shown to be loop free in every steady-state situation. An algorithm is derived which updates routing tables according to changing load conditions. The model incorporates real control and information structures. Lower and upper bounds on cell and blocking are determined. The sequent two algorithm is given which asserts that the network performance objectives can be achieved by means of the state dependent shorts route algorithm.

Keywords: Routing, large scale systems, decentralized control, communication control application, traffic control, estimation and prediction.

Introduction

An analysis and synthesis of traffic management and control in communication networks poses a complex problem. Control rules must be finding for a large set of geographically dispersed local controllers acting under stringent time constraints in a random environment.

Currently, the asynchronous transfer mode is being proposed as an information transfer technique for future integrated service digital network and routing in the network is done by means of routing tables.

During a call the traffic source issues information in bursts. Each burst is sent as a sequence of calls. Characteristics of active and silent periods between calls, bursts, and cells depend on a traffic class. A control system is appended to a node which measures a number of trunks busy on all outgoing links and uses that information to determine a sequence of measurement on a slightly loaded trunk group is:

\[ X(k+1) = aX(k) + bU(k+1) + \xi(k+1), \]

where \( a = \exp(-T) \); \( b = \frac{1}{T}[1 - \exp(-T)]. \) (3)

This an update cycle length normalized by a mean holding time \( T = \Delta t/r \);

Random variables \( X(k), U(k+1), \xi(k+1) \) are Poisson.

The asynchronous transfer mode system is described by the following stochastic processes:

\[ \text{Number of call connections } v_1(t); \text{ number of bursts in progress } v_2(t); \text{ number of cells in the system } X(t) \]

\[ \text{Pr}\{v_1(t) = v_1\} = \sum_{v_1 \in \mathbb{N}} \sum_{v_2 \in \mathbb{N}} \text{Pr}\{v_2(t) = v_2\} \]

\[ \times \text{Pr}\{v_2(t) = v_2, v_3(t) = v_3\} \times \text{Pr}\{v_3(t) = v_3\}. \]

(4)

Denote by \( \hat{D}_m \) the total delay incurred by cells during overload periods in layer \( m \) by \( \hat{v} \) the total number of cell arrivals, define \( \hat{d}_m = \hat{D}_m / \hat{v} \)

and

\[ d = \sum_{m=1}^{M} \hat{d}_m \]

(5)

\[ \hat{d} = \max[\hat{d}_1, ..., \hat{d}_m], \]

(6)

where
The Rower appear bounds on blocking probability \( b \) can be defined in a similar way:

\[
\hat{b} = \sum_{i=1}^{M} b_i ;
\]

and

\[
\tilde{b} = \max\{b_1, b_2, \ldots, b_m\}
\]

with

\[
\hat{b} < b \leq \tilde{b}
\]

Using the process constructed by called the Multilayer Markov chain (MLMC) as a model of the input process give the following structured model of asynchronous transfer mode system:

Number of calls progress

\[
\nu_j(k_1 + 1) = \nu_j(k_2) + A_j(k_1 + 1) - D_j(k_2 + 1);
\]

number of bursts in progress

\[
\nu_j(k_1 + 1) = \nu_j(k_2) + A_j(k_1 + 1) - D_j(k_2 + 1);
\]

number of cell arrivals

\[
\nu_j(k_1 + 1) = A_j(k + 1)
\]

The cells arriving in the lowest layer are put into the buffer with buffer state

\[
X(k_1 + 1) = X(k_1) + \nu_j(k_1 + 1) - Z(k_1 + 1),
\]

Where \( Z(k_1 + 1) \) is the service process (the number of cells transmitted in epoch \( k_1 + 1 \)):

\[
Z(k_1 + 1) = \min\{1; X(k_1) + \nu_j(k_1 + 1)\}
\]

Then that the considered process is stationary and denote by \( \lambda \) an intensity of offered traffic : \( \lambda = \frac{\Theta}{T} \). In the teletraffic engineering \( \lambda \) is measured in Erlands:

\[
\dot{X} = m[X(k)] = \lambda ;
\]

\[
\sigma_x^2 = \text{var}[X(k)] = \lambda ;
\]

\[
\sigma_v^2 = \text{var}[\nu(k)] = T \lambda ;
\]

Aggregated an analytical models

Let us introduce the layer time sculls:

\[
\tau_m = \text{interarrival time of } m \text{ layer entities during an activity period in the } m + 1 \text{ layer};
\]

\[
\tau_m + 1 >> \tau_m >> \tau_{m-1} ;
\]

\[
\nu_m = m - \text{layer entity duration} ;
\]

\[
t = \text{absolute time}; \quad \Delta t = \text{basic observation interval}.
\]

An observation interval \( \Delta t_m \) in layer M is defined by the equation

\[
\Delta t_m = T S_m \Delta t_{m-1},
\]

where \( m = 1, M \), \( \Delta t_0 = \Delta t \), \( TS_m \) defines low many \( m-1 \) layer time units constitute one \( m \) layer time \( k_m, m = 1, k_m \) measures the absolute time in units \( \Delta t_m \) of \( \Delta t_m \):

\[
k_m(t) = \frac{t - t_0}{\Delta t_m}.
\]

Denote by \( u_m(k_m) \) a number of \( m \) layer entities in progress at time \( k_m \), then state of arrival process at time \( t \) is described by the random vector \( U(t) = \{u_m(k_m) : k_m = k_m^\alpha(t), m = 1, M \} \)

If \( m < \bar{m} \) then the mean number of arrivals in layer \( m \) seen in layer \( \bar{m} \) during the time interval \( k_m \), can easily be determined:

\[
A_m(k_m) = \alpha(\bar{m}, m)U_m(k_m); k_m = 1, \bar{m} \quad (22)
\]

where the constant

\[
\alpha(\bar{m}, m) = \frac{\Delta t}{\bar{m} - 1} \prod_{i=m}^{\bar{m}-1} \frac{1}{\tau_i + 1}
\]

The following aggregated mode is arrival process in layer \( \bar{m} \)

\[
U_{\bar{m}}(k_{\bar{m}} + 1) = U_{\bar{m}}(k_{\bar{m}}) + A_{\bar{m}}(k_{\bar{m}} + 1) - D_{\bar{m}}(k_{\bar{m}} + 1),
\]

where \( D_{\bar{m}}(k_{\bar{m}} + 1) \) is a number of departures in layer \( \bar{m} + 1 \) in epoch \( k_{\bar{m}} + 1 \);

arrival process in layer \( \bar{m} + 1 \)

\[
U_{\bar{m} + 1}(k_{\bar{m} + 1}) = U_{\bar{m} + 1}(k_{\bar{m} + 1});
\]

and

\[
X(k_{\bar{m}} + 1) = X(k_{\bar{m}}) + \alpha(\bar{m}, \bar{m})U_{\bar{m}}(k_{\bar{m}} + 1) - Z(k_{\bar{m}} + 1)
\]

Service in process

\[
X(k_{\bar{m}} + 1) = X(k_{\bar{m}}) + \alpha(\bar{m}, \bar{m})U_{\bar{m}}(k_{\bar{m}} + 1) - Z(k_{\bar{m}} + 1)
\]

\[

Service in process

\[
Z(k_{\bar{m}} + 1) = \min\left[ \prod_{i=1}^{\bar{m}} T S_i ; X(k_{\bar{m}}) + \alpha(\bar{m}, \bar{m})U_{\bar{m}}(k_{\bar{m}} + 1) \right] + \alpha(\bar{m}, \bar{m})U_{\bar{m}}(k_{\bar{m}} + 1).
\]
The aggregated models can be implemented as Markov chain simulation. Simulation results for two layers (cell and burst) are shown in Table 1.

Table 1 Numerical and simulation results

<table>
<thead>
<tr>
<th>Nr of channels k</th>
<th>Traffic intensity gi</th>
<th>Mean waiting time [msec]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Event by event simulation</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>0.146</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>0.293</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>0.439</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>0.586</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>0.732</td>
</tr>
<tr>
<td></td>
<td>120</td>
<td>0.878</td>
</tr>
<tr>
<td></td>
<td>140</td>
<td>0.982</td>
</tr>
</tbody>
</table>

Consider a two dimensional Markov Chain (TDMC) (U, X) corresponding to the aggregated model (23÷27). Using the TDMC model it can determine time congestion $S$:

$$S = \sum_{i=1}^{k} \Pi(i, N) \quad (28)$$

and cell blocking probability $b$:

$$b = \sum_{i=1}^{k} \sum_{x=N-i+L+1}^{i} \Pi(i, x), \quad (29)$$

where $(i, x)$ transmissions form state, $j = 0, k$; $(j, k-L+1)$ - states, $\Pi(i, x)$ - probability that the TDMC; The cells started to be blocked when the system enters the region

$$B = \{(i, x): i = L+1, k; x = N-i+L+1, N\}$$

of the state space close to the right boundary. The developed model was used to investigate cell delay and blocking. The impact of particular layers on cell delay in packet voice transmissions is illustrated in Table 2.

Table 2. Cell delay. Three layers.

<table>
<thead>
<tr>
<th>Traffic intensity gi</th>
<th>Three layer simulation</th>
<th>Mean waiting time [msec]</th>
<th>Burst layer d2</th>
<th>Cell layer d1</th>
<th>Total delay d1+ d2+ d3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.586</td>
<td>0.368</td>
<td>0.322</td>
<td>0.004</td>
<td>0.000</td>
<td>0.3225</td>
</tr>
<tr>
<td>0.732</td>
<td>0.996</td>
<td>0.455</td>
<td>0.0161</td>
<td>0.2746</td>
<td>0.7456</td>
</tr>
<tr>
<td>0.878</td>
<td>0.599</td>
<td>1.192</td>
<td>2.9612</td>
<td>62824</td>
<td>632.4</td>
</tr>
<tr>
<td>0.982</td>
<td>18605</td>
<td>4.716</td>
<td>53.313</td>
<td>20510</td>
<td>20510</td>
</tr>
</tbody>
</table>

Denote by $\tilde{r}$ the critical load above which overloads in the burst layer cause long delays, where

$$r = u_2 g_2 \quad \text{or} \quad r = u_3 g_2 g_1.$$ 

Long delays and high cell blocking are avoided when

$$u_3 g_1 g_2 < \tilde{r} \quad (30)$$

or alternatively

$$u_3 g_1 < \tilde{r}.$$ 

They control the flow of entities in different layers:

- cell layer

$$\tilde{g}_1 = \frac{\tilde{r}}{u_3 g_2}; \quad (31)$$

burst layer

$$\tilde{u} = \frac{\tilde{r}}{g_1}; \quad (32)$$

call layer

$$\tilde{u}_3 = \frac{\tilde{r}}{g_1 g_2}. \quad (33)$$

Routing control scheme

Denote by $N$ a set of network nodes: $N = \{(i, j, k, \ldots)\}$, by $L$ a set of network links: $L = \{(i, j, k, l)\}$ and by $R'_{jk} = \{(j, k, l)\}$. An algorithm of updating routing tables consists of the following steps

1. Updating an information $I(k)$ about a state of all links:

$$I(k) = \{x_{jk}(k), u_{jk}(k): (jk) \in L\}. \quad (34)$$

2. Estimate traffic congestion $Z_{jk}(k+1)$ an link $(jk)$ in the approaching time interval:

$$Z_{jk}(k+1) = A_{jk}(I(i), i = 1, n), \forall (j, k) \in L \quad (35)$$

3. Determine qualities $\Omega'_{jk}[x(k)]$ of route $R'_{jk}$;

$$k \in N, k \neq j, \text{ from } j \text{ to } l \text{ for all origin } – \text{destination pairs } j, l \in N.$$ 

4. For each origin $j$ and destination $l$ determine a sequence $\Psi'(k)$ by arranging them according to their quality.

We shall denote by

$$\Psi(k) = \{\Psi'(k): j, l \in N\}. \quad (36)$$

State dependent routing and traffic estimation

Define the random routing variable $\theta'_{jk}$ as a proportion of the input traffic $U'_j$, which is forwarded to the link $(j, k)$.
The above equation define the Model 2 of the network. Assume that the function \( \Phi \) is separable and does not explicitly depend on \( \theta(k) \):

\[
Q = \sum_{k=1}^{k-1} \sum_{(p,q) \in E} \Phi_{pq}(X_{pq}(k+1))
\]

The state equation for the link \((j,k)\) has the form

\[
X_{jk}(k+1) + aX_{jk}(k) + bU_{jk}(k+1) + V_{jk}(k+1), \quad \forall (j,k) \in L, \tag{47}
\]

Under that assumption in order to predict \( x_{jk}(k+1) \), the equation (47) is more useful than the state equations of the Model 2. Taking average in (47) gives

\[
\bar{x}_{jk}(k+1) = ax_{jk}(k) + b\bar{U}_{jk}(k+1), \quad \forall (j,k) \in L.
\]

Routes \( R_{jk}^i \), which are used by the (48) traffic from origin \( j \) to destination \( l \) have same length

\[
\Omega_{jk}^i(k) = \sum_{w_{pq}(k), (j,k) \in \theta(j)} w_{pq}(k), \quad \forall (p,q) \in L
\]

and are not longer than the unused routes. The function \( w_{pq}(k) \) is defined by

\[
w_{pq}(k) = \frac{d\Phi_{pq}(k+1)}{dx_{pq}(k+1)}, \quad (u,v) \in L
\]

and may be interpreted as the state dependent link length. The routing control algorithm consists of avoiding the link saturation which is equivalent to a maximization of the function

\[
\Phi_{av}[X_{av}(k+1)] = [m_{av} - X_{av}(k+1)]
\]

This corresponds to the following form of the route length function for the alternate routes

\[
\Omega_{jk}^i(k) = -[w_{jk}(k) + w_{it}(k)]
\]

For the direct routes we obtain:

\[
\Omega_{jk}^i(k) = -w_{jk}(k), \quad \forall (u,v) \in L
\]

The link length function is given by

\[
w_{av}(k) = m_{av} - x_{av}(k+1), \quad \forall (u,v) \in L
\]

Performance of the proposed algorithm was evaluated by means of simulations. The performance of the two routing algorithms using different route length function is compared in Table 3.
In the case A the function \( \Omega'_{jk}(k) \) had the form (52), (53). In the case B the route length was defined as follows.
\[
\Omega'_{jk}(k) = -\min\{w_{jk}(k), w_{kl}(k)\}, \forall j, k \in N \tag{55}
\]
In both cases the link length \( w_{jk}(k) \) has been calculated according to (54). Since in the investigated network the number of trunks busy was only measured it used the estimate
\[
Z_{jk}(k+1) = x_{jk}(k), \forall (j, k) \in L
\]
of \( x_{jk}(k+1) \) for all values of the update cycle length.

To obtain results presented in Tables we assumed that direct link is always tried as the first choice route, and that only one alternate route can be attempted. The network grade of service was measured as a proportion of lost calls. It seen that for the long update times \( \Delta t \geq 120s \) the algorithm A out performed the algorithm B. The mean holding time \( \tau \) was equal to 240 seconds. It note an interesting fact that the multiple overflow did not improve the efficiency of the investigated network. This is illustrated in Table 4.

### Algorithm 2

1. Take the measurements update the networks state \( x(k) \) and the load \( u(k+1) \). Set \( \theta'_{jk}(k) = 0, \forall j, k, l \in N \).
2. Given \( x(k), u(k+1) \) and the vector \( \theta(k) \) of control variables calculate \( x(k+1) \) and the lengths \( \Omega'_{jk}(k) \) of all possible routes.
3. If there is the unused route \( R_{jk}^l \) which is shorter than the longest used route shift a traffic increment \( \delta \theta'_{jk}(k) \) from that route \( R_{jk}^l \).
   Repeat that for all origin destination pairs \( j \rightarrow l \).
4. If for a given origin destination pair \( f \rightarrow l \) all unused routes are longer than the longest used route than shift the traffic increment \( \delta \theta'_{jk}(k) \) from the longest used route to the shortest used route. Check it for all origin destination pairs \( j \rightarrow l \) which uniquely determine the control variables \( \theta'_{jk}(k) \):
   \[
   Fx(k) + G\theta(k) = H \tag{56}
   \]
   Where \( F \) is the Lagrange function.
   From (56) it is obtain the following feedback control law
   \[
   \theta(k) = -G^{-1}Fx(k) + G^{-1}H \tag{57}
   \]
   The equations (56), (57) can be solved by using the Gauss-Seidel method.
5. Memorize new values \( \theta'_{jk}(k), \forall j, k, l \in N \).
6. If there is at least one source destination pair \( j \rightarrow l \) for which the used routes have different lengths go to the Step2. Otherwise stop. The convergence of this type algorithms was investigated in [1,7].

Assume, however, that in spite of the foregoing difficulties, the control law (57) has been determined substituting from (57) for \( \theta(k) \) in model2 gives
\[
X(k+1) = AX(k) + BU(k+1) + G^{-1}FX(k) + G^{-1}H + V(k+1) \tag{58}
\]
This equation describes a time evolution of the network with state – depended routing.

### Table 3. Network grade of service: % of last calls

<table>
<thead>
<tr>
<th>( \Delta t ) [msec]</th>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1.9615</td>
<td>2.0835</td>
</tr>
<tr>
<td>40</td>
<td>2.0750</td>
<td>2.0750</td>
</tr>
<tr>
<td>60</td>
<td>2.920</td>
<td>2.0580</td>
</tr>
<tr>
<td>120</td>
<td>2.5342</td>
<td>2.5853</td>
</tr>
<tr>
<td>240</td>
<td>2.4747</td>
<td>3.2164</td>
</tr>
</tbody>
</table>

To obtain these results the state protection mechanism was additionally implanted. The state protection consists of prohibiting the use of direct links by overflow traffic of at the instant of measurement the number of free circuits is smaller than the threshold value.

### Table 4. Grade of service of the network with multiple overflows

<table>
<thead>
<tr>
<th>( \Delta t ) [msec]</th>
<th>1</th>
<th>2</th>
<th>Number of overflows</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>1.9474</td>
<td>1.9474</td>
<td>2.075 2.3301 2.6193 2.6533</td>
</tr>
<tr>
<td>40</td>
<td>1.8794</td>
<td>1.9740</td>
<td>2.075 2.3301 2.6193 2.6533</td>
</tr>
<tr>
<td>60</td>
<td>1.9304</td>
<td>1.9474</td>
<td>2.075 2.3301 2.6193 2.6533</td>
</tr>
<tr>
<td>120</td>
<td>2.2366</td>
<td>2.3012</td>
<td>2.567 2.6193 2.6958 2.7043</td>
</tr>
<tr>
<td>240</td>
<td>2.6363</td>
<td>2.7553</td>
<td>2.7213 2.4406 3.1040 3.4272</td>
</tr>
</tbody>
</table>
Conclusions

The structure of arrivals to the asynchronous transfer mode multiplexer has been investigated. The multiplayer Markov Chain model reflecting that structure has been constructed. This was accomplished by breaking down the analysis into layers and considering the discrete time model. The algorithm is based on realistic assumptions about the structure of control and available patterns and system status information. Finding a set of control variables \( \theta(k) \) from (33) is a challenging task as well. From the point of view of network design a method is needed to evaluate traffic statistics form (36).

In the paper the problem of multiservice traffic modeling has been analysed. The structure of arrivals to the asynchronous transfer mode systems multiplexer has been investigated in a Markov chain discrete time model reflecting. It is derived the algorithm which update routing tables in circuit switching networks according to the changing traffic condition. The algorithm is based on realistic assumptions about the structure of control and available system status information.

It is described the network by means of stochastic difference equations and formulated the problem of sequential routing optimization. However, the structure of the two dimensional Markov chain admitted fast matrix multiplication. That property was used to obtain the numerical solution.

References