PAPER IEICE/IEEE Joint Special Issue on Assurance Systems and Networks

Stability and Adaptability of Autonomous Decentralized Flow Control in High-Speed Networks*

Chisa TAKANO†a) *and* **Masaki AIDA**††, ***Regular Members***

SUMMARY This paper focuses on flow control in high-speed networks. Each node in a network handles its local traffic flow on the basis of only the information it is aware of, but it is preferable that the decision-making of each node leads to high performance of the whole network. To this end, we investigate the relationship between the flow control mechanism of each node and network performance. We consider the situation in which the capacity of a link in the network is changed but individual nodes are not aware of this. Then we investigate the stability and adaptability of the network performance, and discuss an appropriate flow control model on the basis of simulation results.

key words: *autonomous decentralized system, flow control, diffusion, feedback*

1. Introduction

In a high-speed network, propagation delay becomes the dominant factor in the transmission delay because the speed of light is an absolute constraint. Therefore, at any given time, a large amount of data is being propagated on links in the network. Figure 1 shows the situation where packets are transmitted in a low-speed/high-speed network. In a low-speed network, a destination node may already receive a first bit of a packet before the local node finishes transmitting all bits of the packet completely. In the high-speed network, on the other hand, there may be the case where most of those packets have not yet reached the destination node in spite of having transmitted many packets from the local node. These packets mean that they are just being transmitted on a link. The amount of such data is characterized by the *bandwidth-delay product*, i.e., the propagation distance multiplied by the transmission rate. Therefore, in high-speed and/or long-distance transmission, there is more data in transit on the links than there is in the nodes [1].

Figure 2 shows an example of how much data there can be on a link. Let us consider the situation involving data transmission between two nodes, a distance of 1 km apart with a link speed of 1 Mbps. If the transmission speed is increased to 1 Gbps, the amount of data on the link is equivalent to that on 10^3 km of a 1-Mbps link. And, if the transmission speed is increased to 1 Tbps, the data volume is equivalent to 10^6 km of a 1-Mbps link. This distance is about 2.5 times the distance between the earth and the moon. Consequently, it is impossible to exert time-sensitive control based on collecting global information about the network. If we allow to spend sufficient time to collect global information, the information so gathered is too old to apply to time-sensitive control. So, in a high-speed network, the frameworks of time-sensitive control are inevitably autonomous decentralized systems [1]–[5].

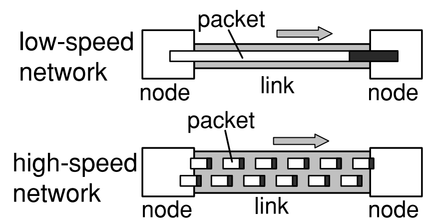

Fig. 1 Effect of large bandwidth-delay product.

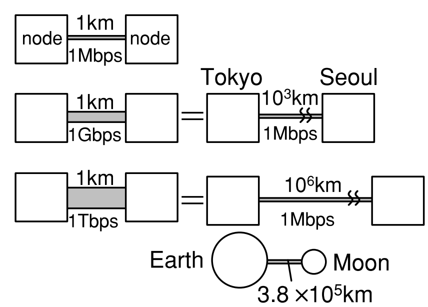

Fig. 2 Example of bandwidth-delay product.

This paper focuses on node-by-node flow control in networks, in which nodes handle their local traffic flow themselves based only on the information they are aware of. It is, of course, preferable that the decision making of each node leads to high performance for the whole network. In flow control, we use the total throughput of a network as a global performance measure [1]. We investigate the behavior of local packet flow and the global performance measure when a node is congested, and discuss an appropriate flow control model on the basis of simulation results. In addition, we

Manuscript received March 7, 2003.
Manuscript revised May 15, 2003.
†The author is with Traffic Engineering Division, NTT Advanced Technology Corporation (NTT-AT), Musashino-shi, 180-0006 Japan.
††The author is with NTT Information Sharing Platform Laboratories, NTT Corporation, Musashino-shi, 180-8585 Japan.
a) E-mail: chisa@m.ieice.org
*A part of this paper was presented at IEEE ICDCS Workshop (ADSN 2002) [2] and at IEEE ISADS 2003 [3].

investigate the stability and adaptability of the network performance when the capacity of a link is changed.

This paper is organized as follows: In Sect. 2, we discuss related works, comparing them with our work by categorizing flow control mechanisms. Section 3 presents a framework for our flow control as an autonomous decentralized system. Section 4 describes two different flow control models based on the framework described in Sect. 3. Section 5 shows the simulation model and conditions. Section 6 shows the simulation results for the performance of the two different flow control models. Finally, we conclude this paper in Sect. 7.

2. Related Works

In general, the mechanism used for flow control for high-speed networks should satisfy the following requirements:

- With regard to the collection of information: it must be possible to collect the information used in the control.
- With regard to the delay in applying control: the control should take effect immediately.

There are many other papers that study optimization of flow control problems in a framework of solving linear programs [6]–[10]. Their works assume the collection of global information about the network, but it is impossible to realize such a centralized control mechanism in high-speed networks. In addition, solving these optimization problems requires enough time to be available for calculation, and so it is difficult to apply them to decision-making in a very short time-scale.

Decentralized flow control by end hosts including TCP is widely used in the current networks, and there is much research in this area [9]–[11]. However, since the end-to-end or the end-to-node control does not apply to decision-making in a time-scale shorter than the round-trip delay, it is insufficient to apply to decision-making in very short time-scale.

In this paper, we propose a simple and effective method of flow control, which satisfies the two above requirements. The concept of our autonomous decentralized control method is based on [4] and our method gives a stable state of the whole network via local decision making at each node. Since the proposed control method uses less information than the control methods described in our previous studies [1], [5], the proposed control is relatively simple.

3. Preliminary Description of Flow Control

3.1 Performance Measure

Each packet in a network is either in a node or on a link. Since the packets currently stored in nodes are not being transmitted over the network, it is natural to

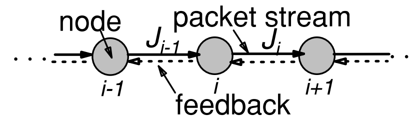

Fig. 3 Interaction between nodes.

define the total throughput of the network as a global performance measure as follows. We define the total throughput of a network at time t as the amount of data being propagated on the network [1]–[5], [12]. In other words, it is the number of packets being propagated on all links in the network at time t.

On the other hand, the only packets we can control are those stored in nodes, and not those being propagated. Thus, higher performance of the whole network involves many uncontrollable packets being propagated on links. Therefore, inappropriate flow control cannot produce a state that has high performance and stability.

3.2 Node Model

Figure 3 shows the interaction of our flow control between nodes using the network model with a simple 1-dimensional configuration. All nodes have two incoming links and two outgoing ones for a one-way packet stream and feedback information, that is, node i ($i = 0, 1, 2, \ldots$) transfers packets to node $i + 1$ and node $i + 1$ sends feedback information (node information) to node i. For simplicity, we assume that packets have a fixed length in bits.

All nodes are capable of receiving and sending node information from/to adjacent downstream and upstream nodes, respectively. Each node i can receive node information sent from the downstream node $i+1$, and can send its node information to the upstream node $i - 1$.

When node i receives node information from downstream node $i + 1$, it determines the transmission rate for packets to the downstream node $i + 1$ using the received node information and adjusts its transmission rate towards the downstream node $i + 1$. The framework of node behavior and flow control is summarized as follows:

- Each node i autonomously determines the transmission rate J_i based only on information it is aware of, i.e., the node information obtained from the downstream node $i + 1$ and its own node information.
- The rule for determining the transmission rate is the same for all nodes.
- Each node i adjusts its transmission rate towards the downstream node $i + 1$ to J_i.
  (If there are no packets in node i, the packet transmission rate is 0.)
- Each node i autonomously creates node informa-

tion according to a predefined rule and sends it to the upstream node $i-1$.
- The rule for creating the node information is the same for all nodes.
- Packets and node information both experience the same propagation delay.

As mentioned above, the framework of our flow control model involves both autonomous decision-making by each node and interaction between adjacent nodes. There is no centralized control mechanism in the network. More precisely, it is impossible to achieve centralized control in a high-speed network environment. Hereafter, we investigate the behavior of the total network performance driven by two different flow control schemes, applied for different processes used to determine the transmission rate.

3.3 Packet Flow

In this paper, we focus on the stability and adaptability of flow control in the congested state, and we consider packet flow in a heavy-traffic environment. The packet flow is defined as the number of sent packets per unit of time, and it is the same as the transmission rate toward the downstream node in a heavy-traffic environment. That is, we let the packet flow be $J_i(t)$ if the transmission rate specified by node i is $J_i(t)$. This is because node i has sufficient packets to transfer. Hereafter, we identify the packet flow with the transmission rate specified by the node.

The packet flow $J_i(t)$ should be controlled by the behavior of node i in the framework described in Sect. 3.2. This means the packet flow can be expressed using the node information obtained from the downstream node $i+1$ and its own node information. We define the packet flow as

$$J_i(t) := \alpha\, r_i(t-d_i) - D\,(n_{i+1}(t-d_i) - n_i(t)), \quad (1)$$

where $n_i(t)$ denotes the number of packets in node i at time t, $r_i(t-d_i)$ is the target transmission rate specified by the downstream node $i+1$ as node information, α (>0) and D (≥ 0) are constants, and d_i denotes the propagation delay between node i and node $i+1$. In addition, $(r_i(t-d_i), n_{i+1}(t-d_i))$ is notified from the downstream node $i+1$ with the propagation delay d_i. The first term on the right hand side of Eq. (1) reflects the target rate specified by the downstream node, and the second term, which is called the diffusion term, is proportional to the gradient of the packet density. We call α and D the flow intensity multiplier and the diffusion coefficient, respectively.

If there is no packet loss in the network, the temporal variation of $n_i(t)$ is expressed as

$$n_i(t+\epsilon) - n_i(t) = \epsilon\left[J_{i-1}(t-d_{i-1}) - J_i(t)\right], \quad (2)$$

where $\epsilon > 0$ is a small number. To estimate the temporal variation roughly, we replace i with x and apply continuous approximation. Then the propagation delay becomes $d_i \to 0$ for all i and the packet flow is expressed as

$$J(x,t) = \alpha\, r(x,t) - D\,\frac{\partial n(x,t)}{\partial x}, \quad (3)$$

and the temporal variation of the number of packets at x is expressed as a diffusion type equation,

$$\frac{\partial n(x,t)}{\partial t} = -\alpha\,\frac{\partial r(x,t)}{\partial x} + D\,\frac{\partial^2 n(x,t)}{\partial x^2}, \quad (4)$$

by using the continuous equation

$$\frac{\partial n(x,t)}{\partial t} + \frac{\partial J(x,t)}{\partial x} = 0. \quad (5)$$

That is, our method aims to perform flow control using the analogy of a diffusion phenomenon. We can expect that packets in the congested node to be distributed to the whole network and normal network conditions to be restored after some time.

Hereafter, we consider two types of flow control and compare them. One type handles the first term and the other controls the first and second terms on the right hand side of Eq. (1).

4. Flow Control Models

4.1 Drift-Type Flow Control

In this subsection, we set $D=0$ in Eqs. (1) and (3), and investigate the characteristics of a flow control scheme whose packet flow is determined only by the first term on the right hand side of Eq. (1).

Let the number of packets in the network be N. To obtain higher network performance, flow control should enable a state in which many packets are being propagated on links. This state corresponds to a state in which there are fewer packets in nodes.

The simplest strategy for achieving this state is for each node to attempt to decrease the number of packets in it. Therefore, the temporal variation of $n_i(t)$ should be

$$n_i(t+\epsilon) - n_i(t) < 0. \quad (6)$$

From Eq. (2) and $D=0$, this strategy means that node i notifies a smaller rate to the upstream node $i-1$ than the rate notified by the downstream node,

$$r_{i-1}(t) < r_i(t-d_i). \quad (7)$$

However, if all nodes use this strategy, then the total throughput decreases with time as a result. Therefore, the strategy described by Eq. (7) cannot be used continuously.

Conversely, if we use the rate specified to the upstream node as

$$r_{i-1}(t) > r_i(t-d_i), \quad (8)$$

then $n_i(t)$ increases with respect to time (when there are many packets in the upstream node). But the buffer in each node has a finite capacity, so this strategy described by Eq. (8) cannot be used continuously either.

If we set the rate specified to the upstream node as

$$r_{i-1}(t) = r_i(t - d_i), \tag{9}$$

then $n_i(t)$ does not change with respect to time under a heavy traffic condition. This means that the strategy described by Eq. (9) does not diminish the total performance of the network. However, when some node is congested, its restoration requires a long time. Thus, the strategy described by Eq. (9) can also not be used continuously.

From the above considerations, we choose the following strategy. The rate specified from node i to the upstream node $i-1$ is determined according to the state of node i. Let the objective of n_i be n_s. If $n_i(t) > n_s$, then $r_{i-1}(t)$ is specified by using Eq. (7); if $n_i(t) < n_s$, then $r_{i-1}(t)$ is specified by using Eq. (8); and if $n_i(t) = n_s$, then $r_{i-1}(t)$ is specified by using Eq. (9).

Since the above flow control uses the value of $n_i(t)$, we call it drift-type flow control in this paper.

4.2 Diffusion-Type Flow Control

In this subsection, we set $D > 0$ in Eqs. (1) and (3), and investigate the characteristics of the flow control scheme whose packet flow is determined by both drift and diffusion terms.

In this control scheme, node i's packet transmission rate to the downstream node $i+1$ is determined as

$$J_i(t) = \alpha\, r_i(t - d_i) - D\,(n_{i+1}(t - d_i) - n_i(t)). \tag{10}$$

In addition, the node information of node i sent to the upstream node $i-1$ is determined as

$$r_{i-1}(t) = J_i(t). \tag{11}$$

In the framework of Eqs. (10) and (11), the node information of i specified to the upstream node $i-1$ is a pair of values $(r_{i-1}(t), n_i(t))$.

In the case where $D = 0$ and $\alpha = 1$, Eqs. (10) and (11) reduce to a drift-type flow control specified by Eq. (9). However, for $D > 0$, since we can control the diffusion term.

Since the above flow control uses the diffusion term, we call it the diffusion-type flow control in this paper.

5. Simulation Model

In this section, we consider a simple network model with a bottleneck link having a narrow bandwidth for comparing the performance of the two different flow control principles described in the previous sections.

5.1 Network Model

Figure 4 shows our network model, which is a closed network with a 1-dimensional configuration and toroidal boundary. The network has a bottleneck link and a corresponding congested node. All the other nodes and links are in the same condition. This model simulates the situation when congestion occurs at a certain node. We are interested in the behavior of the local congestion, that is, whether:

- it causes deterioration of the total network performance through interaction among nodes, or
- it diminishes with time.

Detailed conditions of our network model are listed below.

- Number of nodes: $m = 60$. Each node specified by $i \pmod{60}$.
- Propagation delay between adjacent nodes: 1 (unit time)
- Index of the congested node: $i = 29$
- Total number of packets in the network: $N = 6000$
- Maximum number of packets on a link (except the bottleneck link): $L_c = 100$
- Maximum number of packets on the bottleneck link (between nodes $i = 29$ and 30): $L_b = 10$, 25, 50, or 75
(that is, 1/10, 1/4, 1/2, or 3/4 of the bandwidth of other links, and the same length)

To investigate the stability under congestion, in addition to the above conditions, we set the initial condition for congested node $i = 29$ as follows.

- Number of packets in node $i = 29$ at time $t = 0$: 400
- The other 5600 packets are randomly configured in other nodes and on other links.

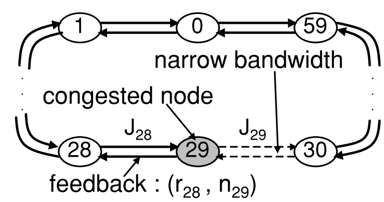

Fig. 4 Network model with a bottleneck link.

5.2 Drift- and Diffusion-Type Flow Control Schemes

As a model for the drift-type flow control, we set an objective for the number of packets in a node to be $n_s = 60$, and set the following transmission rate and node information.

$$J_i(t) = \min(r_i(t), L_i), \tag{12}$$

$$r_{i-1}(t) = \begin{cases} J_i(t) - L_i/10 & (n_i(t) > n_s), \\ J_i(t) & (n_i(t) = n_s), \\ J_i(t) + L_i/10 & (n_i(t) < n_s), \end{cases} \tag{13}$$

where L_i denotes the link capacity between nodes i and $i+1$.

As a model for the diffusion-type flow control, Eqs. (10) and (11), we use the following flow control model. Since the packet flow is restricted by the link capacity, the diffusion-type flow control is expressed as follows:

$$J_i(t) = \min(\max(\tilde{J}_i(t), 0), L_i), \tag{14}$$

$$r_{i-1}(t) = J_i(t), \tag{15}$$

where

$$\tilde{J}_i(t) = \alpha\, r_i(t - d_i) - D\,(n_{i+1}(t - d_i) - n_i(t)) \tag{16}$$

$$L_i = \begin{cases} L_b, & (i = 29), \\ L_c, & (\text{otherwise}), \end{cases} \tag{17}$$

$$\alpha = 1.00 \;\text{ or }\; 1.01, \tag{18}$$

$$D = 0.1. \tag{19}$$

The value of α (> 1) determines the increase speed of the average rate of the packet flow. If the value of α becomes larger, recovery time of throughput will become much less when the capacity of a bottleneck link is recovered. We want to investigate whether the total throughput is recovered or not, even if the difference between the value of α and 1 is little, so we use the above two values of α as the simulation model. The value of D determines the speed of smoothing the distribution of the number of packets. If D takes a larger value, then the packet flow of the whole network will smooth quickly, in an ideal situation. However, in an actual situation, if we set a large value of D, the packet flow is frequently restricted by the capacity of the link through min operation in Eq. (14). In this case, diffusion phenomena do not appear in packet flow behaviors. We set comparatively small D, since we expect that behaviors of the packet flow simulate diffusion phenomena.

6. Simulation Results: Stability and Adaptability

From the simulation results for the drift- and diffusion-type control models, we compare the total throughput of the network. In addition, we discuss the stability and adaptability of the both types of flow control model through the observation of the total throughput.

6.1 Stability and Adaptability in the Case of the Appearance of a Bottleneck Link

This subsection considers the case where the capacity of a link in the network is suddenly reduced to a narrow bandwidth. No node is aware of the change of the link state and new capacity of the link. We investigate the stability and adaptability of the drift- and diffusion-type control models through the observation of the total throughput of the network.

Figures 5, 6 and 7 show the total throughput for drift- and diffusion-type (with $\alpha = 1.00$ and 1.01) flow control models, respectively. The horizontal axis denotes the simulation time and the vertical axis denotes the total throughput (i.e., the total number of packet being propagated on links). Each line in these figures

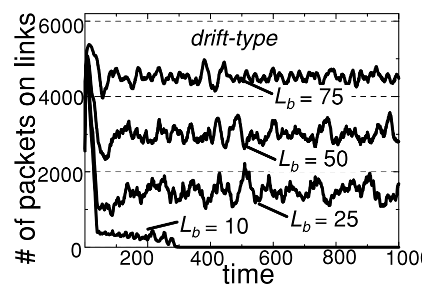

Fig. 5 Temporal behavior and stability of the total throughput of the network for the drift-type flow control scheme.

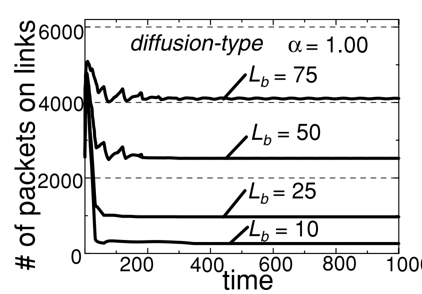

Fig. 6 Temporal behavior and stability of the total throughput of the network for the diffusion-type flow control scheme ($\alpha = 1.00$, $D = 0.1$).

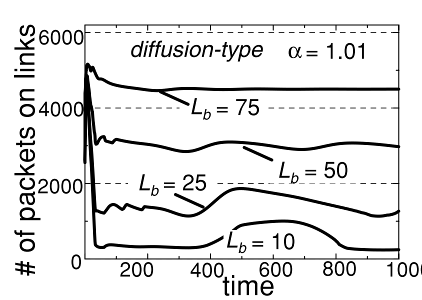

Fig. 7 Temporal behavior and stability of the total throughput of the network for the diffusion-type flow control scheme ($\alpha = 1.01$, $D = 0.1$).

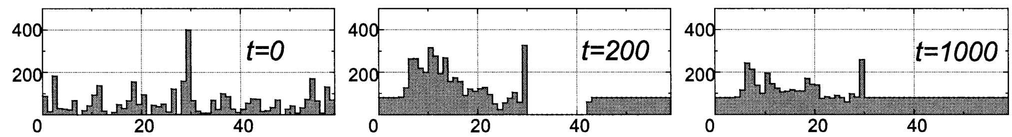

Fig. 8 Distribution of the number of packets stored in each node for the drift-type flow control scheme ($L_b = 10$, $n_s = 60$).

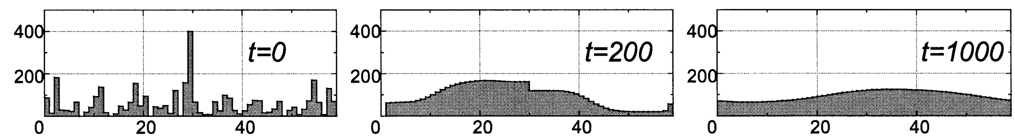

Fig. 9 Distribution of the number of packets stored in each node for the diffusion-type flow control scheme ($L_b = 10$, $\alpha = 1.0$, $D = 0.1$).

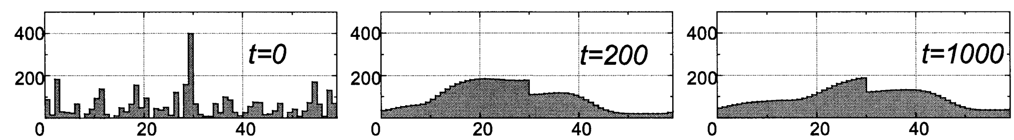

Fig. 10 Distribution of the number of packets stored in each node for the diffusion-type flow control scheme ($L_b = 10$, $\alpha = 1.01$, $D = 0.1$).

shows the result in the case of the capacity of the bottleneck link, $L_b = 10$, 25, 50, or 75.

Figure 5 shows the drift-type flow control scheme achieves high total throughput in the cases where $L_b =$ 25, 50 and 75. However, in the other case where $L_b =$ 10, i.e., the capacity of the bottleneck link is small, the drift-type fails to control the total throughput, which falls to zero with time.

From Figs. 6 and 7, on the other hand, the diffusion-type flow control schemes achieve stable total throughput of the network. It is remarkable that stability is achieved irrespective of the value of L_b. The diffusion-type with $\alpha = 1.01$ achieves higher total throughput than that obtained from the diffusion-type with $\alpha = 1.00$ for all the values of L_b.

We discuss the results from a quantitative point of view. Let us compare two types of flow control schemes in the case where $L_b = 10$, i.e., 1/10 of the other link capacities. For the drift-type control model, the total throughput decreases with time. This means that the flow control model inappropriately influences the global performance of the network. For the diffusion-type control models, on the other hand, the total throughput decreases with time but becomes stable around 270 (for $\alpha = 1.00$) and 600 (for $\alpha = 1.01$). From the link capacity of the bottleneck link L_b, the maximum value of the sustainable total throughput (the maximum number of packets being propagated stably on links) is 600, i.e., 10 packets/link $\times$ 60 links. Thus, the diffusion-type flow control model with $\alpha = 1.00$ achieves only 45% of the maximum value of the total throughput, but that with $\alpha = 1.01$ causes the almost 100% total throughput.

Next, in the case where the total throughput is decreasing, we investigate the movement of packets in the network model.

Figure 8 shows the simulation result for the drift-type flow control model when $L_b = 10$. The horizontal axis of each graph denotes node ID and the vertical axis denotes the number of packets stored in the node. In addition, t denotes the simulation time and initially $t =$ 0. For the drift-type flow control model, the number of packets in the congested node $i = 29$ decreases with time, but and the number of packets stored in each node is uneven at $t = 1000$.

Similarly, Figs. 9 and 10 show the simulation results for the diffusion-type flow control model under the same conditions as described for Fig. 8. We chose parameters as $\alpha = 1.00$ and 1.01, respectively, and $D = 0.1$. In both figures, the number of packets in congested node $i = 29$ decreases with time. For the diffusion flow controls with both $\alpha = 1.00$ and $\alpha = 1.01$, the distribution of the number of packets stored in nodes is smoothly distributed over the network at $t = 1000$.

If we can choose an appropriate value of the objective n_s for the drift-type flow control, the total throughput may be stable and adaptive. The case where $L_b = 25$, 50 or 75 implies that this is feasible. However, the value of n_s should depend on the bandwidth of the bottleneck link as recognized from Fig. 5. Since nodes cannot be aware of information about the bandwidth in a high-speed network environment, the drift-type control cannot achieve high performance. On the other hand, in the diffusion-type control models both for $\alpha = 1.00$ and 1.01, although no node is aware of the

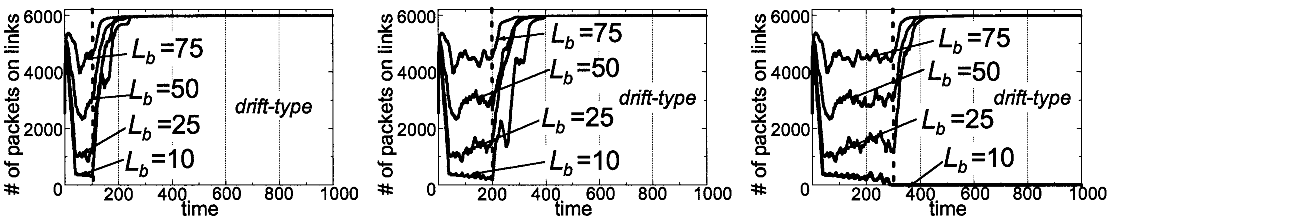

Fig. 11 Temporal behavior and stability of the total throughput of the network for the drift-type flow control scheme and three different times at which the bottleneck is restored.

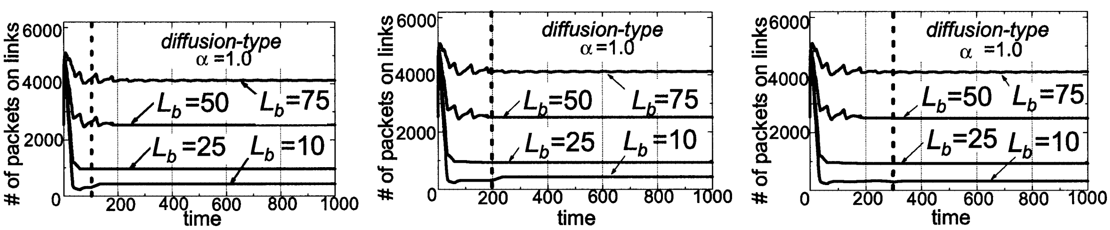

Fig. 12 Temporal behavior and stability of the total throughput of the network for the diffusion-type flow control scheme ($\alpha = 1.00$, $D = 0.1$) and three different times at which the bottleneck is restored.

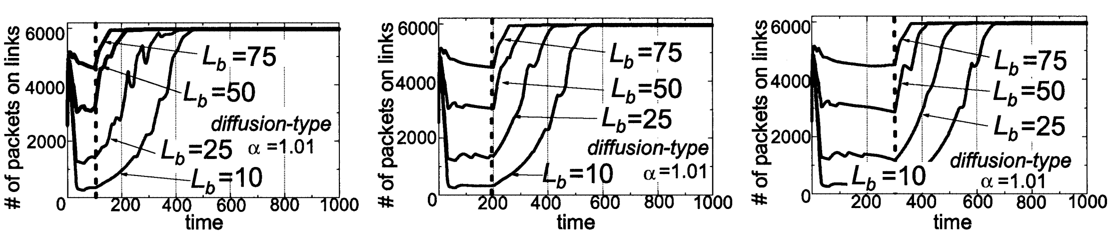

Fig. 13 Temporal behavior and stability of the total throughput of the network for the diffusion-type flow control scheme ($\alpha = 1.01$, $D = 0.1$) and three different times at which the bottleneck is restored.

bandwidth of the bottleneck link, stable performance is achieved. Especially, the diffusion-type control model with $\alpha = 1.01$ provides the higher total throughput than that with $\alpha = 1.00$.

6.2 Stability and Adaptability in the Case of Restoration of a Bottleneck Link

This subsection considers the situation where the capacity of the bottleneck link is suddenly restored. No node is aware of the change of the link state and the restored capacity of the link. We investigate the stability and adaptability of both types of control through observation of the total throughput of the network.

Figures 11, 12 and 13 show the total throughput in the case where the capacity of the bottleneck link L_b is restored to 100 at time $t = 100$, 200, and 300 for these models. These figures show the results for drift-type and diffusion-types with $\alpha = 1.00$ and 1.01. The horizontal axis denotes the simulation time and the vertical axis denotes the total throughput. A broken line indicates the time when the capacity of the bottleneck l ink is restored. The simulation conditions are the same as the case for the previous subsection except for the restoration of the bottleneck link. We show the cases of four different capacities of the bottleneck link, $L_b = 10$, 25, 50, and 75.

In the cases where $L_b = 25$, 50 and 75, the drift-type flow control restores high total throughput of around 6000 as shown in Fig. 11. It is independent of the time when the restoration occurs. However, in the case where $L_b = 10$, i.e., the initial capacity of the bottleneck link is small, for the drift-type flow control the total throughput falls to zero with time. There is no change in the total throughput, even if the capacity of the bottleneck link is restored at time $t = 300$.

On the other hand, the diffusion-type flow control with $\alpha = 1.00$ achieves stable total throughput of the network in Fig. 12. However, although the capacity of the bottleneck link is restored, the total throughput is not restored in any of the cases. This is because we set the flow intensity multiplier of $\alpha = 1.00$ for the diffusion-type flow control. This setting of the flow intensity multiplier is derived from Eq. (9) for balancing input and output flows.

Figure 13 shows that the diffusion-type flow control

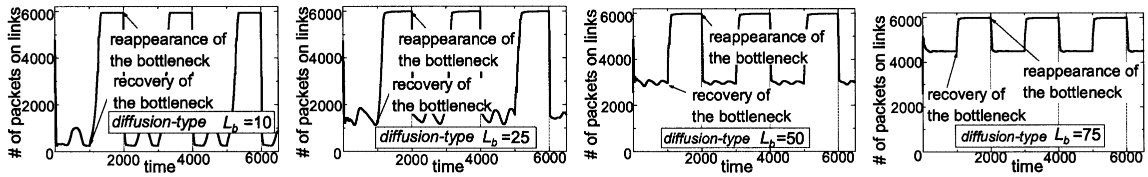

Fig. 14 Temporal behavior and stability of the total throughput for the diffusion-type flow control scheme ($\alpha = 1.01$, $D = 0.1$) when the capacity of the bottleneck is alternately restricted and restored.

with $\alpha = 1.01$ achieves stable total throughput of the network and it is adaptively restored in all of the cases when the link capacity is restored. This characteristic is came from the flow intensity multiplier of $\alpha > 1$.

6.3 Stability and Adaptability in the Case where a Bottleneck Appears and Recovers Repeatedly

This subsection considers the situation where the capacity of the bottleneck link alternates between normal and restricted values. No node is aware of the changes in the link state. We investigate the stability and adaptability of the diffusion-type control with the flow intensity multiplier of $\alpha > 1$ through observation of the temporal behavior of the total throughput of the network.

Figure 14 shows the total throughput in case where the bottleneck appears and recovers repeatedly for the diffusion-type control with $\alpha = 1.01$. The capacities of the bottleneck links are $L_b = 10$, 25, 50, and 75, respectively. The horizontal axis denotes the simulation time and the vertical axis denotes the total throughput.

The period of the bottleneck link being alternately changed between appearance and restoration is 1000. Simulation conditions are the same as the case for the previous subsection except for the state of the bottleneck link. In all cases, the diffusion-type control with $\alpha = 1.01$ achieves the adaptable global performance of the network.

7. Conclusions

This paper has presented a framework for flow control in high-speed networks as an autonomous decentralized system. We have showed two typical node-by-node flow control models based on the framework. The drift-type flow control depends on the number of packets in a node and the diffusion-type flow control depends on the gradient of the number of packets in a node. For both types of control, nodes handle their local traffic flow themselves based only on the information they are aware of.

To investigate the behavior of local packet flow and the global performance measure when a node is congested, we compared two types of the models through simulations. For comparison, we used the total throughput as the flow control performance measure.

Although the drift-type control cannot achieve high performance adaptively, the diffusion-type does achieve stable performance in congested situations. In particular, the diffusion-type control with the flow intensity multiplier of $\alpha > 1$ achieves high performance adaptively, even in a situation in which the congested state changes dynamically.

We are interested in the appropriate values of the flow intensity multiplier and the diffusion coefficients, α and D. These issues will be the subject of further study.

References

[1] M. Aida and K. Horikawa, "Stability analysis for global performance of flow control in high-speed networks based on statistical physics," IEICE Trans. Commun., vol.E82-B, no.12, pp.2095–2106, Dec. 1999.
[2] M. Aida and C. Takano, "Stability of autonomous decentralized flow control schemes in high-speed networks," Proc. IEEE ICDCS 2002 Workshop (ADSN 2002), pp.63–68, July 2002.
[3] C. Takano and M. Aida, "Stability and adaptability of autonomous decentralized flow control," Proc. 6th International Symposium on Autonomous Decentralized Systems (ISADS 2003), pp.311–318, Pisa, Italy, April 2003.
[4] M. Aida and C. Takano, "Principle of autonomous decentralized flow control and layered structure of network control with respect to time scales," Supplement of the ISADS 2003 Conf. Fast Abstracts, pp.3–4, Pisa, Italy, April 2003.
[5] K. Horikawa, M. Aida, and T. Sugawara, "Traffic control scheme under the communication delay of high-speed networks," Proc. Int. Conf. on Multi-Agent Systems (ICMAS'96), pp.111–117, Dec. 1996.
[6] Y. Bartal, J. Byers, and D. Raz, "Global optimization using local information with applications to flow control," Proc. 38th Ann. IEEE Symp. on Foundations of Computer Science, Oct. 1997.
[7] K. Kar, S. Sarkar, and L. Tassiulas, "A simple rate control algorithm for maximizing total user utility," Proc. IEEE INFOCOM 2001, pp.133–141, 2001.
[8] S.H. Low and D.E. Lapsley, "Optimization flow control-I: Basic algorithm and convergence," IEEE/ACM Trans. Netw., vol.7, no.6, pp.861–874, 1999.
[9] S. Kunniyur and R. Srikant, "A decentralized adaptive ECN marking algorithm," Proc. IEEE GLOBECOM 2000, pp.1719–1723, 2000.
[10] J. Mo and J. Walrand, "Fair end-to-end window based congestion control," IEEE/ACM Trans. Netw., vol.8, no.5, pp.556–567, Oct. 1999.
[11] R. Johari and D.K.H. Tan, "End-to-end congestion control for the internet: Delays and stability," IEEE/ACM Trans. Netw., vol.9, no.6, pp.818–832, 2001.

[12] D. Mitra, "Optimal design of windows for high speed data networks," Proc. IEEE INFOCOM'90, pp.1156–1163, 1990.

Chisa Takano received the B.E. degree in Telecommunication Engineering from Osaka University, Japan, in 2000. In 2000 she joined Traffic Research Center, NTT Advanced Technology Corporation (NTT-AT). She has been engaged in research and development of computer networks. She received the Young Investigators Award of IEICE in 2003.

Masaki Aida received his B.Sc. and M.Sc. degrees in Theoretical Physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, and received the Ph.D. degree in Telecommunication Engineering from the University of Tokyo, Japan, in 1999. Since joining NTT Laboratories in 1989, he has been mainly engaged in research on traffic issues in ATM networks and computer communication networks. From March 1998 to March 2001, he was a manager at Traffic Research Center, NTT Advanced Technology Corporation (NTT-AT). He is currently a Senior Research Engineer at NTT Information Sharing Platform Laboratories. His current interests include traffic issues in communication systems. He received the Young Investigators Award of IEICE in 1996. Dr. Aida is a member of the IEEE and the Operations Research Society of Japan.