# Study on Compatibility of Diffusion-Type Flow Control and TCP

Kaori Muranaka*, Chisa Takano*, Masaki Aida**

*NTT Advanced Technology Corporation,
2-4-15, Naka-cho, Musashino-shi, 180-0006 Japan.
TEL: +81 422 37 0813 FAX: +81 422 60 2387
E-mail: {kaori.muranaka, chisa.takano}@ntt-at.co.jp

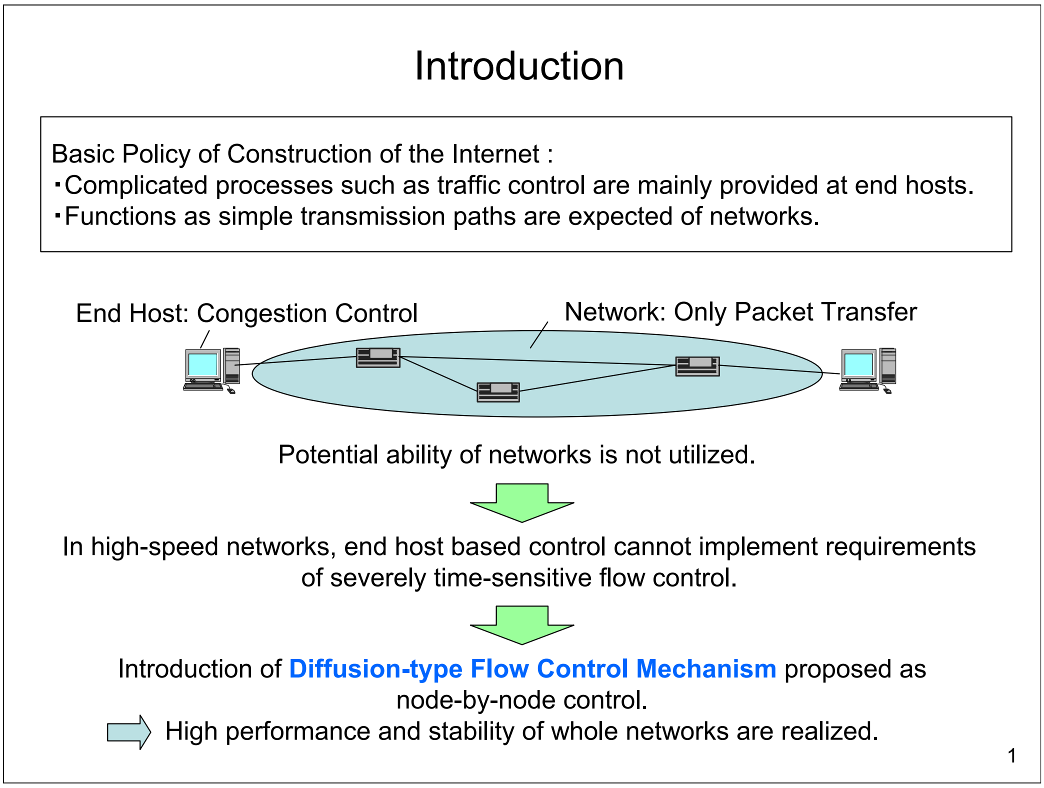
**Tokyo Metropolitan University,
Hino-shi, 191-0065 Japan.
TEL/FAX: +81 42 585 8627
E-mail: maida@cc.tmit.ac.jp

## Abstract

We have proposed the diffusion-type flow control mechanism as a solution of severely time-sensitive flow control required for high-speed networks. In this mechanism, each node in a network manages its local traffic flow on the basis of only the local information directly available to it, by using predetermined rules. In the current IP based networks, end-to-end flow control including TCP is widely used. However, end-to-end control cannot be applied to decision-making in a time-scale shorter than the round-trip delay, since end hosts provide the flow control. In this paper, we investigate the performance of diffusion-type flow control and TCP by using the simulation tool ns2. Then, we show that diffusion-type flow control can coexist with TCP and can be complementary to each other.

**Keywords**: flow control, diffusion, TCP, autonomous decentralized control

**Contact Person**: Kaori Muranaka (kaori.muranaka@ntt-at.co.jp)



## Introduction

Basic Policy of Construction of the Internet :
- Complicated processes such as traffic control are mainly provided at end hosts.
- Functions as simple transmission paths are expected of networks.

End Host: Congestion Control

Network: Only Packet Transfer

Potential ability of networks is not utilized.

In high-speed networks, end host based control cannot implement requirements of severely time-sensitive flow control.

Introduction of **Diffusion-type Flow Control Mechanism** proposed as node-by-node control.

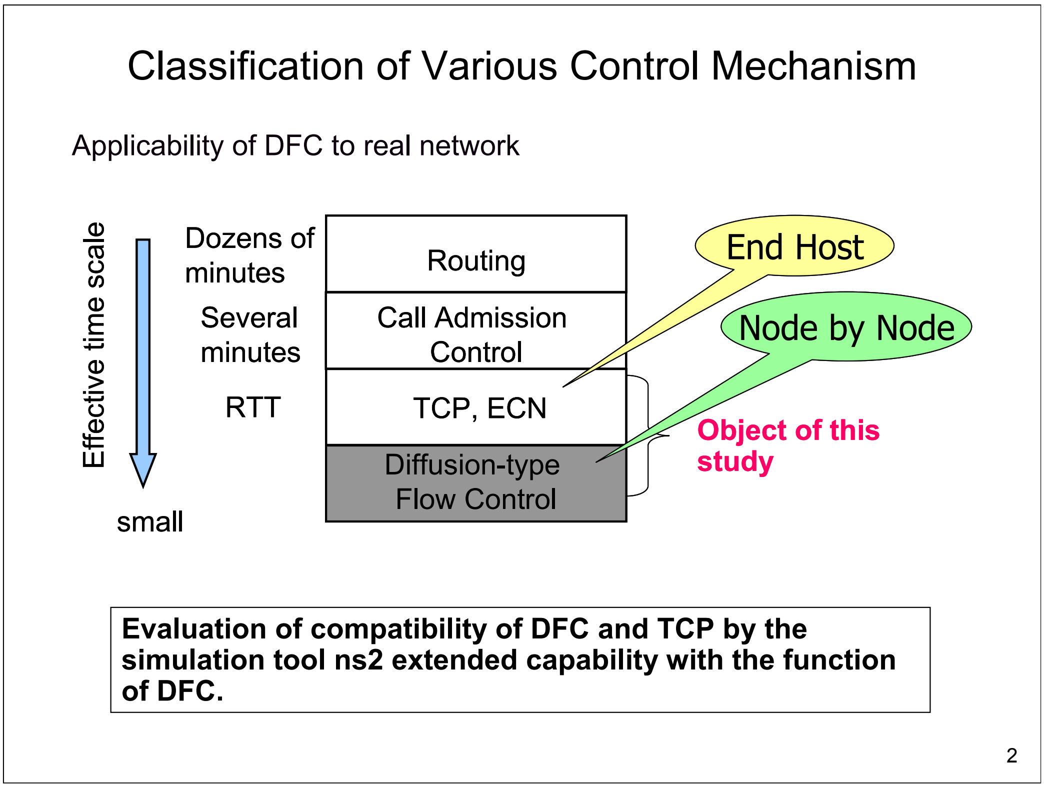
⇨ High performance and stability of whole networks are realized.

## 1. Introduction

The rapid spread of the Internet will necessitate the construction of higher-speed backbone networks in the near future. In a high-speed network, data transfer process at a node speeds up, so a large number of packets are in transit on links in the network. This situation increases sensitivity of network performance. This is because control delay greatly affects network performance. Therefore, the severely time-sensitive traffic control is necessary for high-speed networks.

Basic policy to construct current IP-based networks is as follows. Complicated processes such as congestion control are assigned to end-hosts. Nodes in a network deal with only process of packet transfer. To achieve severely time-sensitive traffic control, node-by-node control is preferable with respect to a small control delay. In current networks, end-host based traffic control is widely used, such as end-to-end flow control including TCP [1], [2]. This control cannot utilize potential ability of networks.

We have proposed a diffusion-type flow control (DFC) mechanism [3]-[5]. This control mechanism is a solution for severely time-sensitive flow control that is required for high-speed networks. We have evaluated the performance of DFC and verified that DFC provides high and stable performance in a whole network.



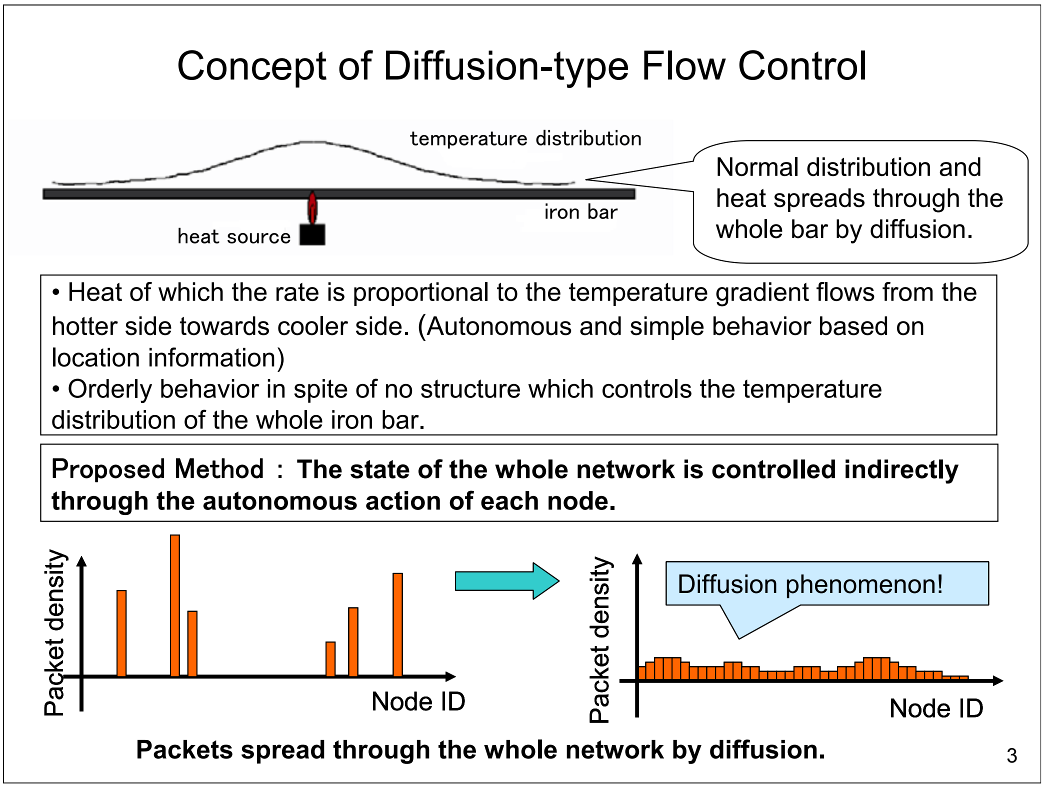
## Classification of Various Control Mechanism

Applicability of DFC to real network

**Evaluation of compatibility of DFC and TCP by the simulation tool ns2 extended capability with the function of DFC.**

In order to apply DFC to actual networks, we should verify the compatibility and complementarity of DFC with existing control mechanisms in networks. Various control mechanisms can be classified from the point of view of their particular time scale of control operations. The above figure shows the mutual relationship of different types of control according to such a classification. They form a layered structure for the time scales. For example, routing and call admission control fall into the long and the medium time scales, respectively. Individual mechanisms work well for their appropriate time scales and they cooperate with each other. An end-to-end control such as TCP acts on the time scale of the roundtrip time (RTT). In high-speed networks, since a lot of packets are in transit on links, even the delay such as RTT greatly affects the network performance. However, since end hosts provide the flow control, TCP cannot be applied to decision-making in a time scale shorter than the RTT. The target of DFC is a time scale shorter than the RTT.

In this paper, we investigate the compatibility of DFC and TCP by the simulation tool ns2 [6] extended capability with the function of DFC.



## Concept of Diffusion-type Flow Control

Normal distribution and heat spreads through the whole bar by diffusion.

- Heat of which the rate is proportional to the temperature gradient flows from the hotter side towards cooler side. (Autonomous and simple behavior based on location information)
- Orderly behavior in spite of no structure which controls the temperature distribution of the whole iron bar.

**Proposed Method : The state of the whole network is controlled indirectly through the autonomous action of each node.**

Diffusion phenomenon!

**Packets spread through the whole network by diffusion.**

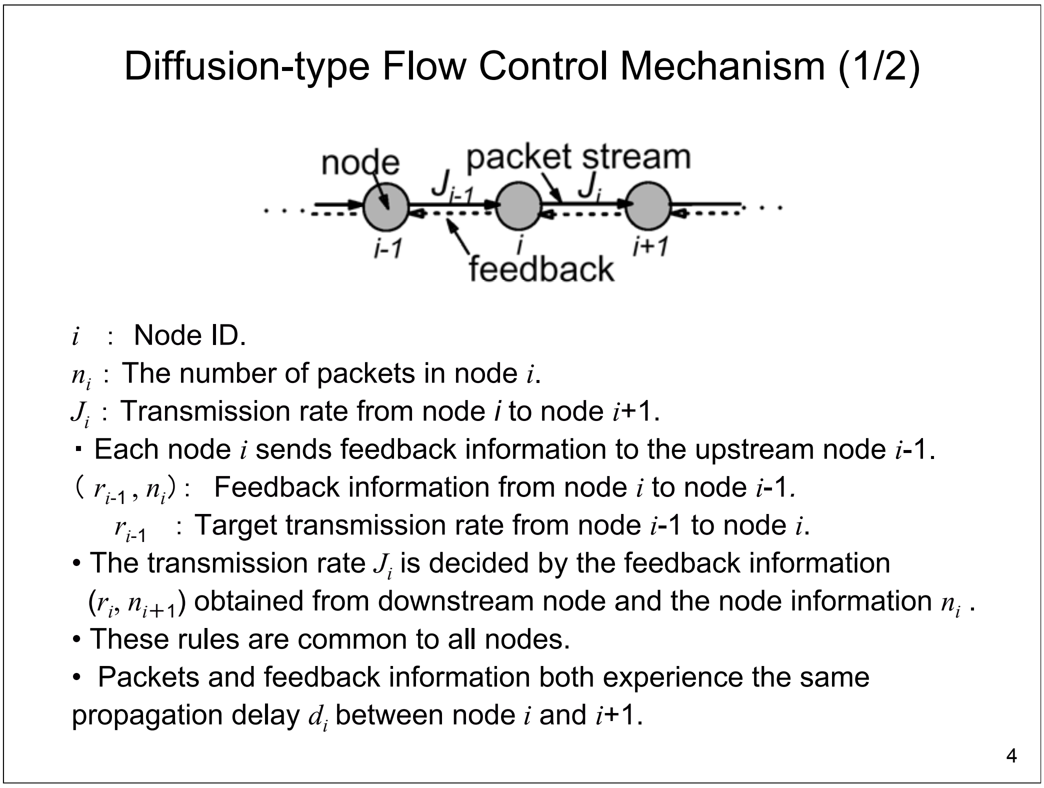
## 2. Diffusion-type Flow Control

### 2.1 Concept

DFC provides a framework in which the implementation of the decision-making of each node leads to high performance for the whole network. The principle of our model can be explained through the following analogy [5].

When we heat a point on a cold iron bar, the temperature distribution follows a normal distribution and heat spreads through the whole bar by diffusion. In this process, the action in a minute segment of the iron bar is very simple: heat flows from the hotter side towards cooler side. The rate of heat flow is proportional to the temperature gradient. There is no direct communication between two distant segments of the iron bar. Although each segment acts autonomously, based on its local information, the temperature distribution of the whole iron bar exhibits orderly behavior.

In DFC, each node controls its local packet flow. A part of the packet rate consists of the rate that is proportional to the difference between the number of packets in the node and that in an adjacent node. Thus, the distribution of the total number of packets in a node in the network becomes uniform over time. In this control mechanism, the state of the whole network is controlled indirectly through the autonomous action of each node.



## Diffusion-type Flow Control Mechanism (1/2)

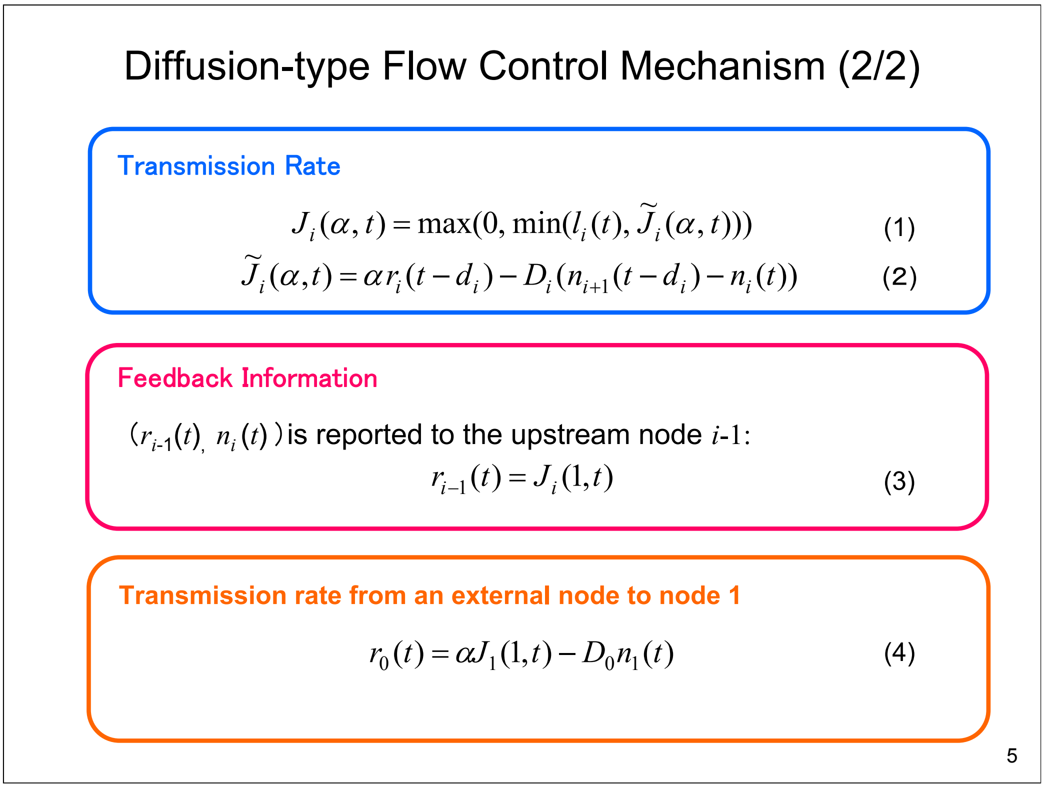
$i$ : Node ID.
$n_i$ : The number of packets in node $i$.
$J_i$ : Transmission rate from node $i$ to node $i$+1.
・Each node $i$ sends feedback information to the upstream node $i$-1.
( $r_{i-1}$ , $n_i$) : Feedback information from node $i$ to node $i$-1.
  $r_{i-1}$ : Target transmission rate from node $i$-1 to node $i$.

- The transmission rate $J_i$ is decided by the feedback information ($r_i$, $n_{i+1}$) obtained from downstream node and the node information $n_i$ .
- These rules are common to all nodes.
- Packets and feedback information both experience the same propagation delay $d_i$ between node $i$ and $i$+1.

### 2.2 Diffusion-type Flow Control Mechanism

We describe the mechanism of DFC [3]-[5]. In DFC, each node controls its local packet flow autonomously. The above figure shows the interactions between nodes in our flow control method, using a network model with a simple 1-dimensional configuration. Each node $i$ transfers packets to the downstream node $i$+1. Each node $i$ can receive feedback information sent from the downstream node $i$+1 and can send feedback information about itself to the upstream node $i$-1. When node $i$ receives feedback information from the downstream node $i$+1, it determines the transmission rate for packets to the downstream node $i$+1 by using the received feedback information, and it adjusts its transmission rate towards the downstream node $i$+1.

$n_i$ and $J_i$ denote the number of packets in node $i$ and the transmission rate from node $i$ to the downstream node $i$+1, respectively. ($r_{i-1}$, $n_i$) is feedback information created by node $i$, where $r_{i-1}$ is the target transmission rate from node $i$-1 to the node $i$. Each node $i$ decides the transmission rate $J_i$ to the downstream node $i$+1 by using feedback information reported from node $i$+1 and the number of packets in node $i$. The transmission rate is decided every time feedback information is received. These rules are common to all nodes. Packets and feedback information both experience the same propagation delay $d_i$ between node $i$ and $i$+1. Each node $i$+1 sends feedback information to the upstream node $i$ at the interval time that is proportional to the propagation delay $d_i$.



## Diffusion-type Flow Control Mechanism (2/2)

**Transmission Rate**

$$J_i(\alpha,t)=\max(0,\min(l_i(t),\tilde{J}_i(\alpha,t))) \qquad (1)$$

$$\tilde{J}_i(\alpha,t)=\alpha r_i(t-d_i)-D_i(n_{i+1}(t-d_i)-n_i(t)) \qquad (2)$$

**Feedback Information**

$(r_{i\text{-}1}(t), n_i(t))$ is reported to the upstream node $i$-1:

$$r_{i-1}(t)=J_i(1,t) \qquad (3)$$

**Transmission rate from an external node to node 1**

$$r_0(t)=\alpha J_1(1,t)-D_0 n_1(t) \qquad (4)$$

We explain the details of DFC. The transmission rate $J_i$ of node $i$ at time $t$ is determined by the above equations (1) and (2), where $l_i$(t) denotes the value of the available bandwidth of the link from node $i$ to node $i$+1 for target flow at time $t$, $n_i$(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 feedback information.

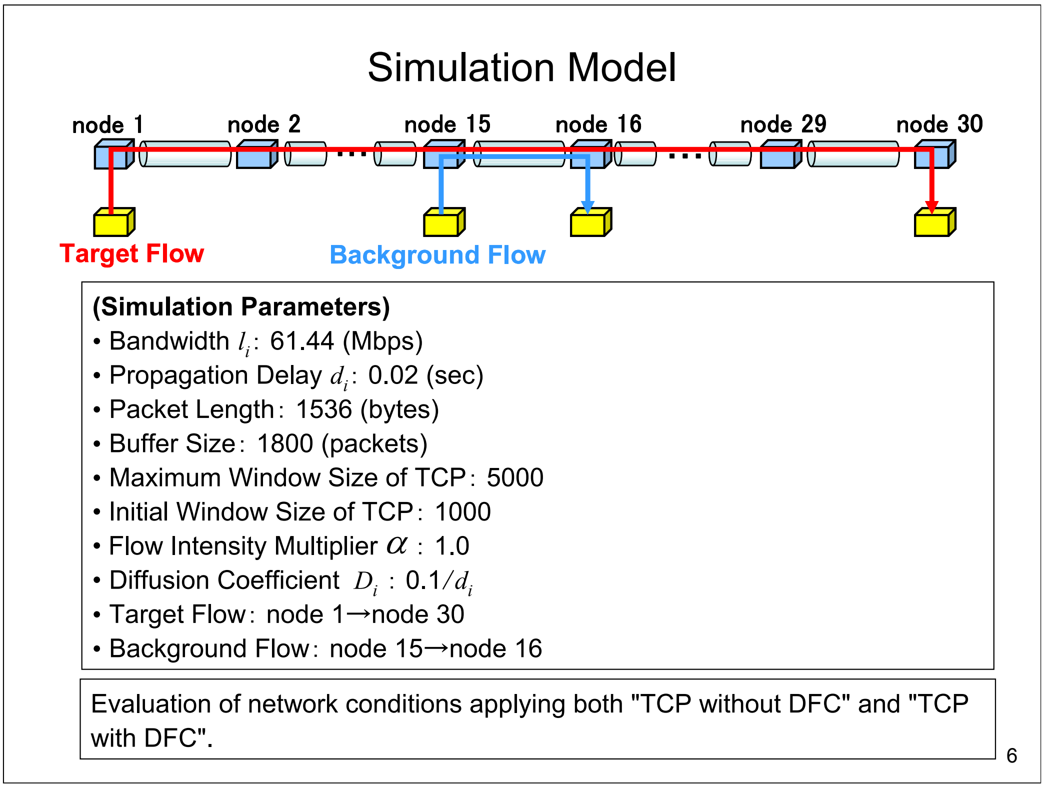
In addition, $r_i(t- d_i)$ and $n_{i+1}(t- d_i)$ are reported from the downstream node $i$+1 as feedback information with propagation delay $d_i$. Parameter $\alpha$ (>=1), which is a constant, is the flow intensity multiplier. Parameter $D_i$ is chosen to be inversely proportional to the propagation delay [4] as follows:

$$D_i=\frac{D}{d_i}\propto(d_i)^{-1}$$

where $D$ (>0), which is a positive constant, is the diffusion coefficient.

The feedback information ($r_{i\text{-}1}$, $n_i$) is reported to the upstream node $i$-1. Here, the target transmission rate is determined as the equation (3).

Moreover we consider a rule for determining $r_0$ as a boundary condition. Node 1 can calculate $J_0$ if we assume that the number of packets stored in the node of the other network is $i$ = 0. The target rate $r_0$, reported by node 1, is created as $\tilde{J}_0$ with the above assumption. That is, it is expressed as the equation (4). This quantity can be calculated just from information known to node 1.



## Simulation Model

**(Simulation Parameters)**

- Bandwidth $l_i$: 61.44 (Mbps)
- Propagation Delay $d_i$: 0.02 (sec)
- Packet Length: 1536 (bytes)
- Buffer Size: 1800 (packets)
- Maximum Window Size of TCP: 5000
- Initial Window Size of TCP: 1000
- Flow Intensity Multiplier $\alpha$ : 1.0
- Diffusion Coefficient $D_i$ : $0.1/d_i$
- Target Flow: node 1→node 30
- Background Flow: node 15→node 16

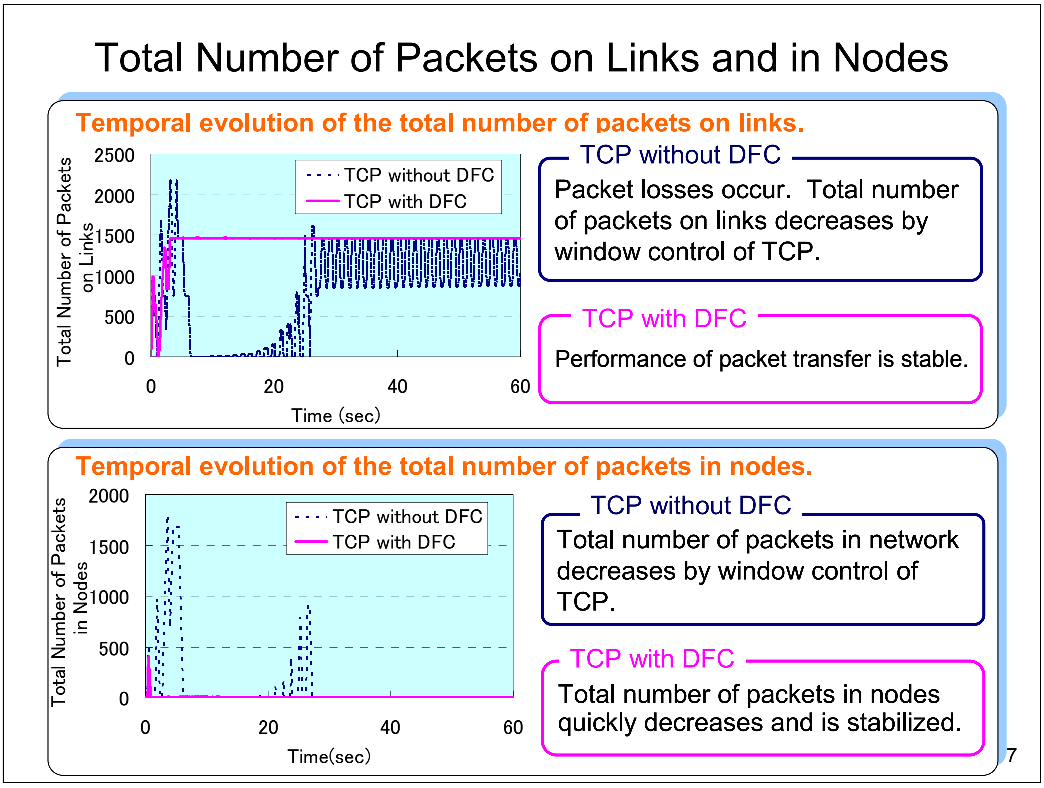
Evaluation of network conditions applying both "TCP without DFC" and "TCP with DFC".

## 3. Evaluations of Compatibility of DFC with TCP

### 3.1 Simulation Model

If DFC is implemented in actual networks, it operates with TCP flow control. So, we should verify the compatibility and complementarity of DFC with TCP. We extend the simulation tool ns2 capability with the function of DFC to investigate the performance of DFC combined with TCP.

The above figure shows our network model with 30 nodes, which is used in the simulations. It represents a part of a network and describes a path of the end-to-end flow extracted from the whole network. Propagation delay and bandwidth of each link between nodes are 0.02sec and 61.44Mbps, respectively. If the packet length is 1536 bytes, the transfer rate of each link is 100 packets/sec. The buffer size of each node is 1800 packets. The flow intensity multiplier and the diffusion coefficient are 1.0 and $0.1/di$, respectively. The target flow is between node 1 and node 30. The background flow is between node 15 and node 16. We investigate the network performance in case of TCP flow control without DFC and TCP combined with DFC.



## Total Number of Packets on Links and in Nodes

**Temporal evolution of the total number of packets on links.**

TCP without DFC

Packet losses occur. Total number of packets on links decreases by window control of TCP.

TCP with DFC

Performance of packet transfer is stable.

**Temporal evolution of the total number of packets in nodes.**

TCP without DFC

Total number of packets in network decreases by window control of TCP.

TCP with DFC

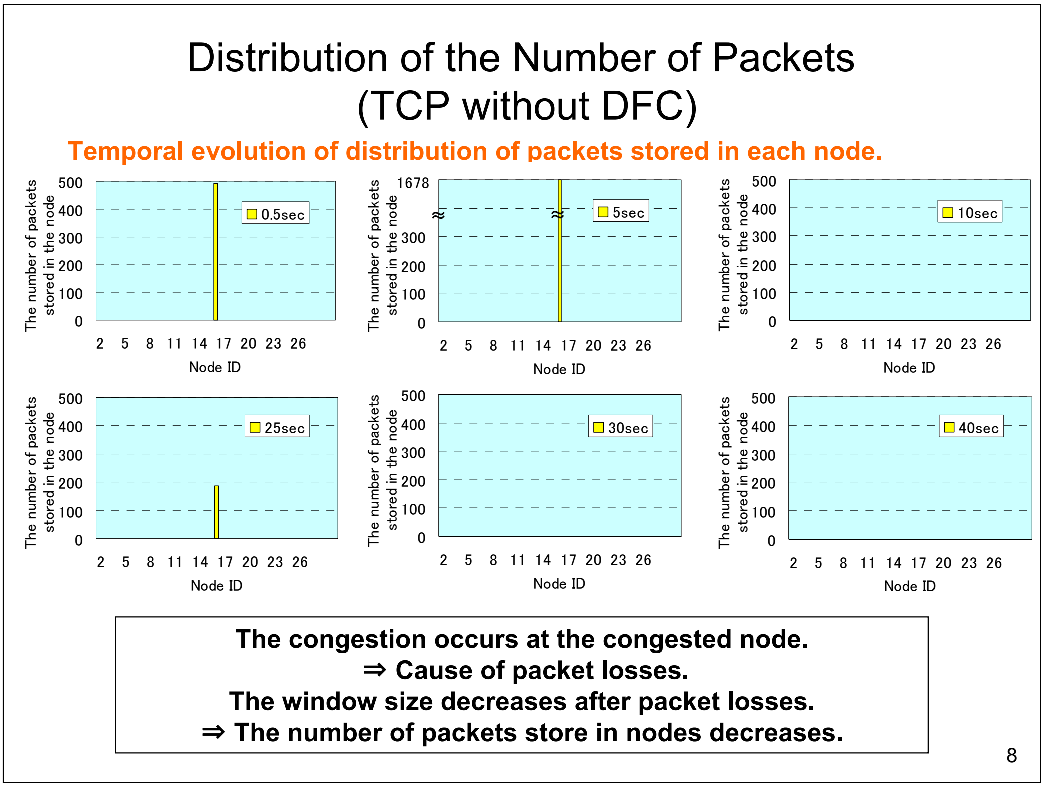
Total number of packets in nodes quickly decreases and is stabilized.

### 3.2 Total Number of Packets on Links and in Nodes

The figure at the top shows temporal evolution of the total number of packets on links for the target flow in case of TCP without DFC and TCP with DFC. The horizontal axis denotes the simulation time and the vertical axis denotes the total number of packets on links. The total number of packets on links indicates the effectiveness of packet transmission about the whole network. If this value is about 1500 ((50 packets/link) * 29 links) packets in our simulation, the network works effectively.

From this figure for TCP without DFC, after the total number of packets on links decreases, it increases by degree and oscillates around the value of 1500 packets that is maximum. Packet losses occur at the congestion node 15, so packets that flow into the network are constrained by the reduction of the TCP window size. The reason for the oscillation is that the processes to transfer packets for the TCP window size and to wait the ack packets are repeated. The packets are not transferred for waiting time of the ack packets, so the total number of packets on links is reduced for a certain period. On the other hand, the behavior for TCP with DFC is stable around the value of 1500 packets.

The figure at the bottom shows temporal evolution of the total number of packets in nodes for the target flow in case of TCP without DFC and TCP with DFC. The horizontal axis denotes the simulation time and the vertical axis denotes the total number of packets stored in nodes. We can see from this figure that the total number of packets in nodes quickly decreases and is stabilized for TCP with DFC .



Distribution of the Number of Packets
(TCP without DFC)

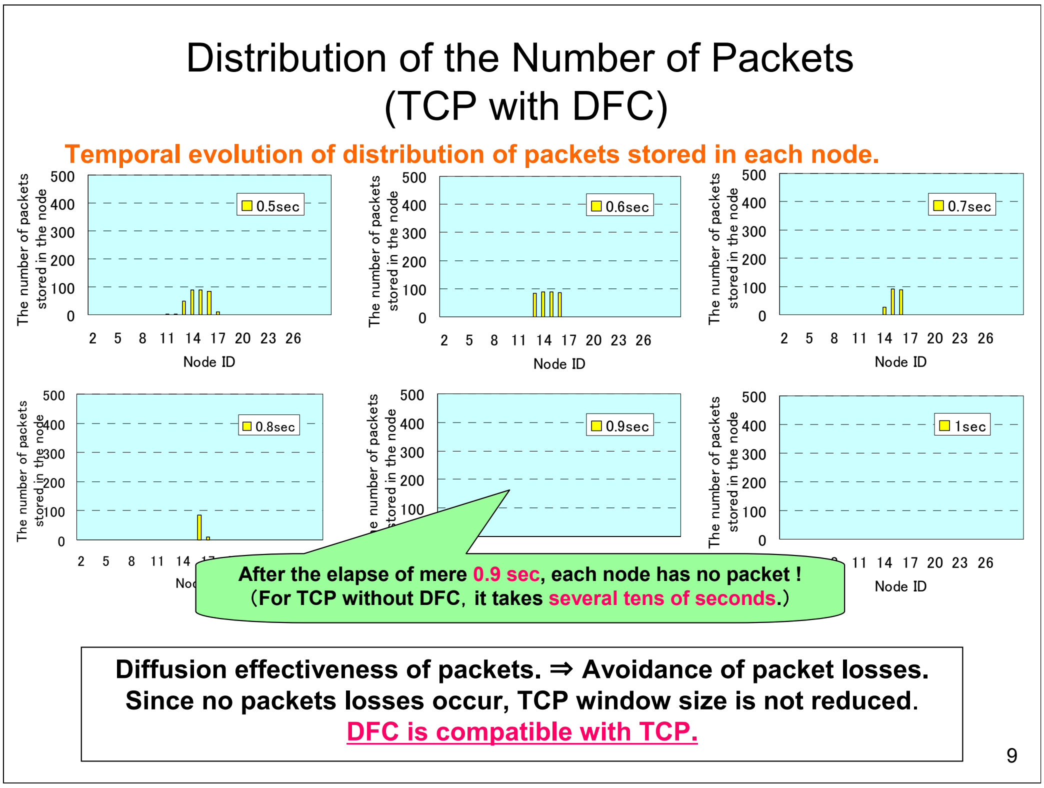
Temporal evolution of distribution of packets stored in each node.

**The congestion occurs at the congested node.**
**⇒ Cause of packet losses.**
**The window size decreases after packet losses.**
**⇒ The number of packets store in nodes decreases.**

### 3.3 Distribution of the Number of Packets for TCP without DFC

We investigate the temporal evolution of distribution of packets stored in each node in case of TCP without DFC. These figures show the result of the investigation, where the horizontal axes denote the node ID and the vertical axes denote the number of packets stored in the node. The value in each figure denotes the simulation time *t*.

We can see the congestion occurs at node 15. This congestion leads to packet losses. Since the window size decreases after packet losses, the number of packets stored in nodes is reduced.



## Distribution of the Number of Packets (TCP with DFC)

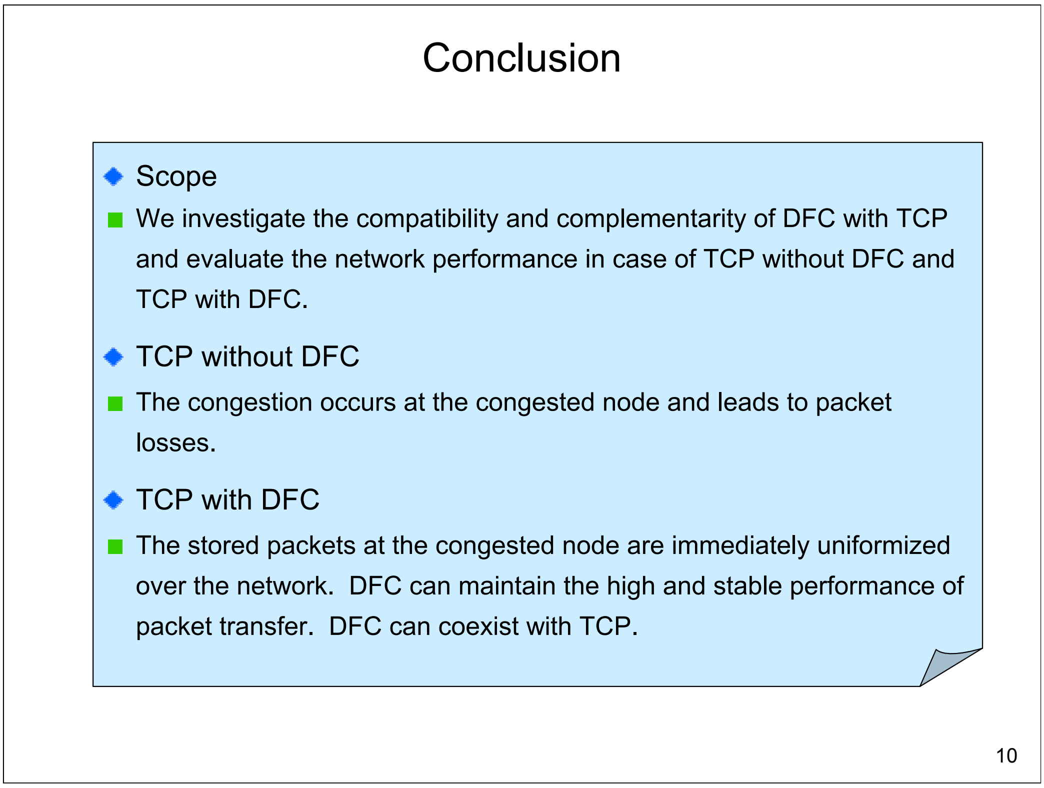
**Temporal evolution of distribution of packets stored in each node.**

**After the elapse of mere 0.9 sec, each node has no packet !**
**(For TCP without DFC, it takes several tens of seconds.)**

**Diffusion effectiveness of packets. ⇒ Avoidance of packet losses.**
**Since no packets losses occur, TCP window size is not reduced.**
**DFC is compatible with TCP.**

### 3.4 Distribution of the Number of Packets for TCP with DFC

Similarly, we investigate in case of TCP with DFC. From these figures, the stored packets do not be concentrated at the congestion node 15. The stored packets at node 15 are immediately uniformized over the network. Thus, nodes in the network cooperatively work in order to avoid packet losses. In addition, the total number of packets stored in nodes decreases rapidly and falls to zero with time. These effects result from the DFC mechanism. Since no losses occur, TCP window size is not reduced. From these result, we conclude DFC is compatible with TCP.



## Conclusion

- Scope
  - We investigate the compatibility and complementarity of DFC with TCP and evaluate the network performance in case of TCP without DFC and TCP with DFC.
- TCP without DFC
  - The congestion occurs at the congested node and leads to packet losses.
- TCP with DFC
  - The stored packets at the congested node are immediately uniformized over the network. DFC can maintain the high and stable performance of packet transfer. DFC can coexist with TCP.

## 4. Conclusion

To apply DFC to actual networks, we should verify the compatibility and complementarity of DFC with TCP that is widely used as end-to-end flow control in networks. We extended the simulation tools ns2 capability with the function of DFC and investigate the performances of TCP without DFC and TCP with DFC. From the investigation, we can obtain the result that the performances of network are high and stable in case of TCP with DFC compared with TCP without DFC. This result means DFC can coexist with TCP.

In the future, we will investigate the compatibility of DFC with more complex types of TCP algorithms such as Vegas and Reno and evaluate the performance of DFC in condition of complex traffic streams.

### References

[1] J. Mo and J. Walrand, “Fair end-to-end window based congestion control,” IEEE/ACM Trans. Networking, vol.8, no.5 pp.556-567, Oct. 1999.

[2] R. Johari and D. Tan, “End-to-end congestion control for the Internet: Delays and stability,” IEEE/ACM Trans. Networking, vol.9, no.6, pp.818-832, Dec. 2001.

[3] C. Takano and M. Aida, “Stability and adaptability of autonomous decentralized flow control in high-speed networks,” IEICE Transactions on Communications, vol.E86-B, no.10, pp.2882-2890, 2003.

[4] C. Takano and M. Aida, and S. Kuribayashi, “Autonomous decentralized flow control in high-speed networks with inhomogeneous configurations,” IEICE Transactions on Communications, vol.E87-B, no.6, pp.1551-1560, 2004.

[5] C. Takano and M. Aida, “Diffusion-type autonomous decentralized flow control for end-to-end flow in high-speed networks,” IEICE Transactions on Communications, vol.E88-B, no.4, 2005.

[6] http://www.isi.edu/nsnam/ns/