A Novel Software-Based H.323 Gateway with Proxy-TC for VoIP Systems

Presenter : Wei-Sheng Yin
Advisor : Dr. Po-Ning Chen

Institute of Communications Engineering
National Chiao Tung University

Agenda

- **Introduction**
- **Motivation**
- **Proposed System Architecture**
  - System Architecture
  - Proxy-Transcoder (Proxy-TC)
  - The GW-TC Signaling
  - GW-TC Signaling Procedure
  - The Handoff Mechanism
  - The Error Recovery Procedure
  - Data Structures
- **Simulation Model and Results**
- **Conclusions**

Introduction

- Due to the constant growth of transmission bandwidth, real-time applications such as VoIP gradually become feasible on the Internet.
- VoIP can bring us advantages like toll-by-pass, coexistence of voice and data services, and easy deployment of telephony network between two branch offices.
- In techniques, VoIP carries voice traffic as *data packets* over a *packet-switched network*, instead of as a synchronous stream of binary sampled data over a circuit-switched, time-division multiplexed (TDM) network.

Introduction - *VoIP Protocols*

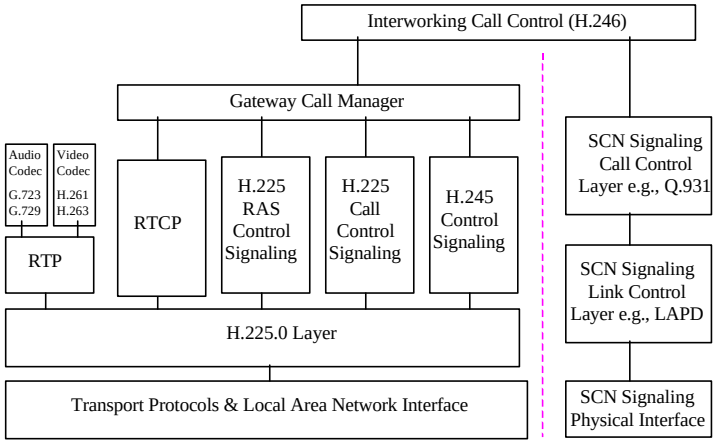
- Features of the VoIP protocols
  - Signaling, call control, and media transport
- Existing protocols for VoIP
  - Recommendation H.323 *by ITU-T*
  - Session Initiation Protocol and Session Description Protocol (SIP/SDP) *by IETF*
  - Media Gateway Control Protocol (MGCP) *by IETF*
  - Media Gateway Control (MEGACO) *by IETF*
    - Been submitted to ITU-T SG-16 for decision as Recommendation H.248
- Of the protocols above, H.323 is perhaps the most widely deployed; hence, our design is based on H.323 protocol stack.

Introduction - *H.323 components*

- **Gatekeeper**
  - Manage a zone (i.e., a collection of H.323 devices). Required Functionality: Address translation, admissions control and bandwidth control.
- **H.323 Terminal**
  - An endpoint on a LAN. Support real-time, 2-way communications with another H.323 entity. Must support voice (audio codec) and signaling (Q.931, H.245, RAS).
- **Gateway**
  - Provide interoperability between different networks, and convert signaling and media. E.g., IP/PSTN gateway.
- Multi-point controller (MC), multi-point processor (MP) and Multi-point control Unit (MCU) are used to support conference calls.

Network Technology Lab

Introduction – Gateway protocol stack



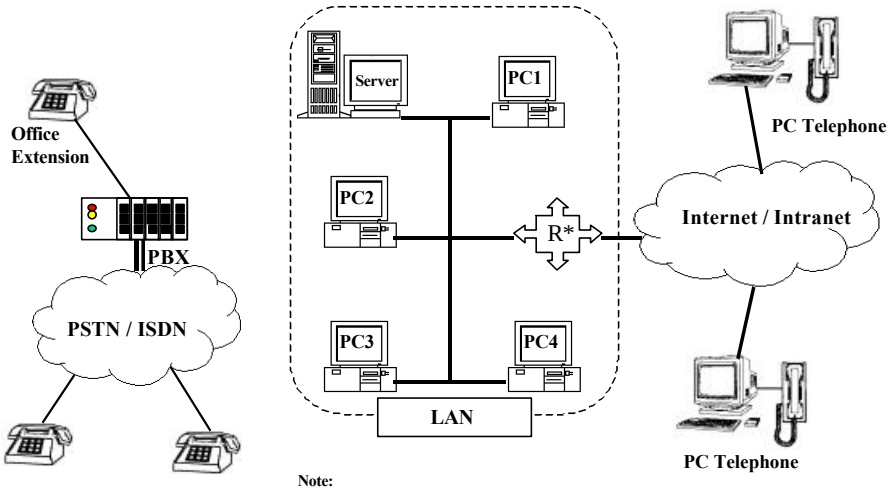
Network Technology Lab

Motivations

- The telephony links maintained by a software-based gateway at times suffer short interrupts due to the possible uneven switching of PC platforms among several computation-bound tasks.
- We observe that the computation powers of the desktop PCs in use for general employees are sometimes not in full utilization.
- This leads us to the idea of gathering these excessive computation powers to share the load of the software-based gateway, and hence, reduce the number of computation-bound tasks on it.
- In such case, one may further increase the gateway capacity without introducing additional hardware cost.

Network Technology Lab

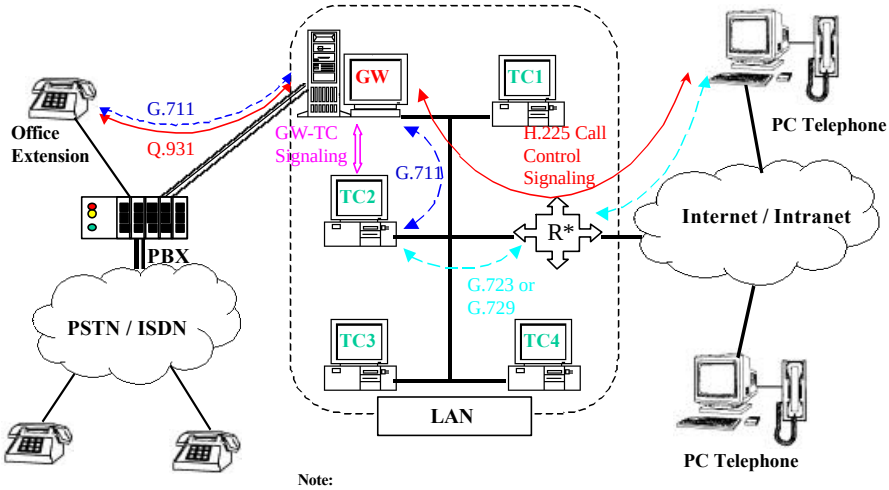
Proposed System Architecture



Note:
*The edge router must support QoS functionality.

Network Technology Lab

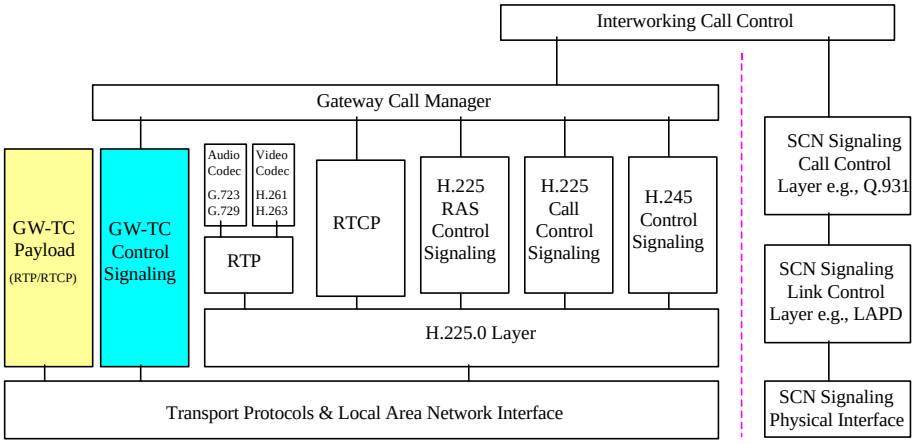
Proposed System Architecture



Note:
*The edge router must support QoS functionality.

Network Technology Lab

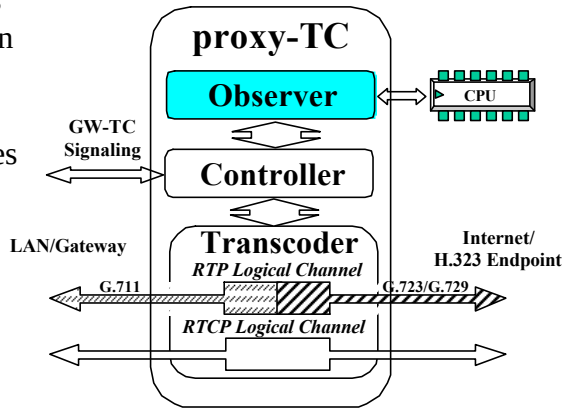
Proposed System Architecture
- Modified Gateway Protocol Stack



Network Technology Lab

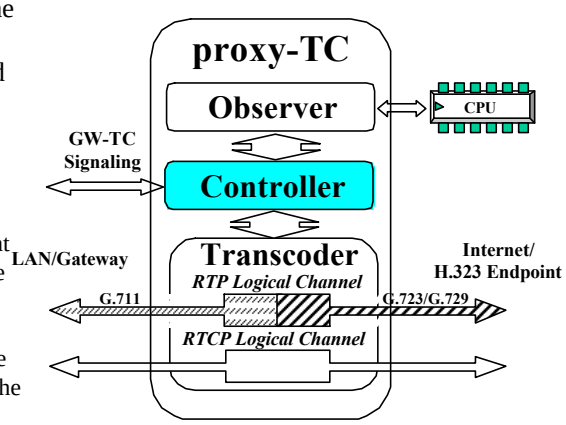
Proposed System Architecture
- Proxy-Transcoder (Proxy-TC)

- The Observer monitors local resource condition on PCs, such as available CPU power, and periodically updates the information for use by Controller.



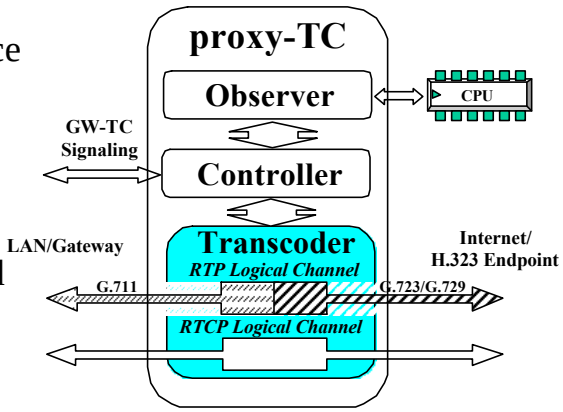
Proposed System Architecture
- Proxy-Transcoder (Proxy-TC)

- The Controller maintains the communication between a proxy-TC and its associated gateway by using GW-TC signaling procedures.
- These control messages include:
  - (1) informing the gateway that the proxy-TC is still active upon the request of its associated gateway, receiving the request of the transcoding service from the gateway; and
  - (3) updating the information of the CPU utilization.



Proposed System Architecture
- Proxy-Transcoder (Proxy-TC)

❖ The Transcoder is responsible for voice stream transcoding between different speech coding algorithms, such as G.711, G.723.1 and G.729.



Proposed System Architecture
- The GW-TC Signaling (1/2)

- The major goal of the GW-TC signaling is to enable the cooperation between the gateway and the proxy-TCs. There are six commands in the GW-TC signaling, which are described below:
  - **Register:** The proxy-TCs send this command manually to the gateway (through web-browsing activity) to register to the gateway so that the gateway can create corresponding proxy-TC record.
  - **Setup:** This command is used by the gateway to request one of the proxy-TCs to perform the transcoding service upon the occurrence of a new call.
  - **Release:** The gateway uses this command to release the transcoding service either (1) upon the end of a call or (2) upon the receipt of the **Emergency** command from a proxy-TC.

Proposed System Architecture
- The GW-TC Signaling (2/2)

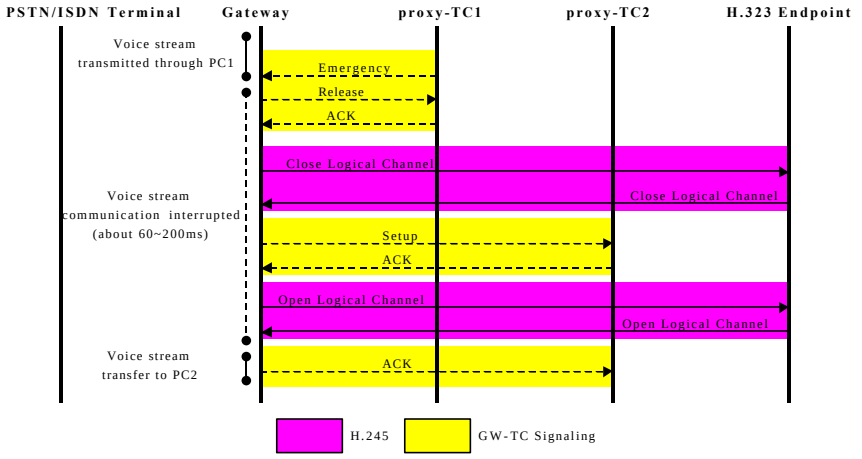
- **Update:** Proxy-TCs use this command to update their latest statuses kept on the gateway.
- **Query:** If the update timer expires the gateway will send **Query** command to request the proxy-TC to update its status .
- **Emergency:** This command is sent by proxy-TCs to notify the gateway of their (sudden) shortage of transcoding resources.
- **Acknowledge (ACK):** This command may be sent by either the gateway or proxy-TCs to inform the other side that the previous command has been successfully received.

Proposed System Architecture
- The GW-TC Signaling Procedure

- Normal call setup procedure @
- Normal call clearing procedure

Proposed System Architecture

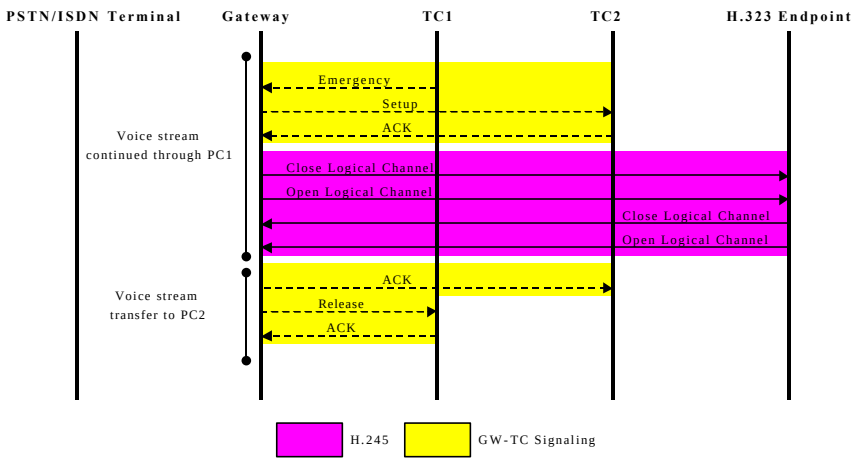
- The Handoff Mechanism – Hard Handoff



Network Technology Lab

Proposed System Architecture

- The Handoff Mechanism – Soft Handoff



Network Technology Lab

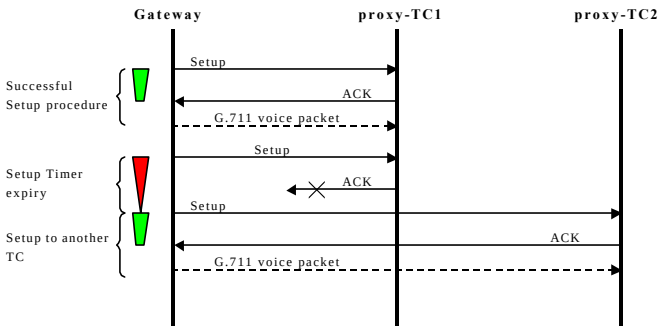
Proposed System Architecture

- Error Recovery Procedure

- Situations concerned
  - Setup
  - Update
  - Emergency
  - Release

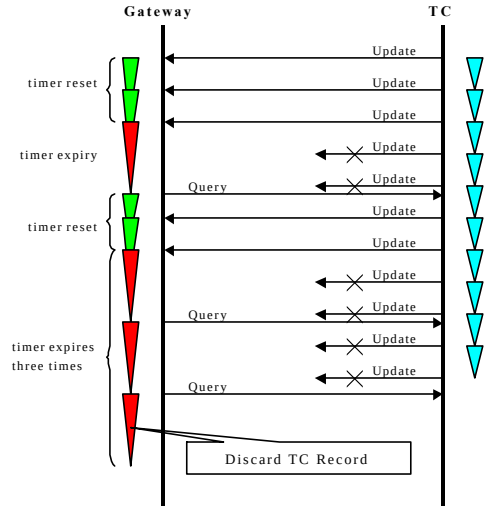
Proposed System Architecture

- Error Recovery Procedure -- Setup



Proposed System Architecture

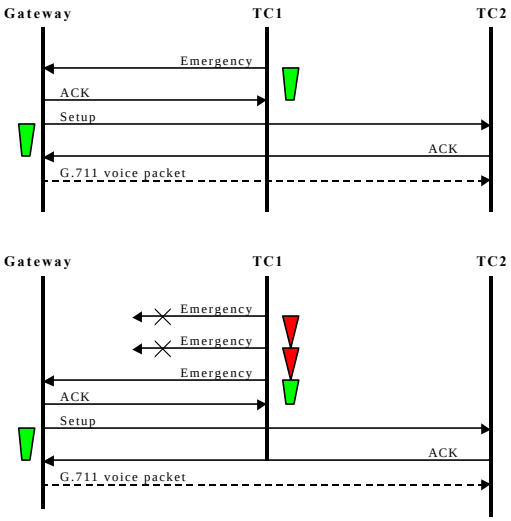
- Error Recovery Procedure -- Update



Network Technology Lab

Proposed System Architecture

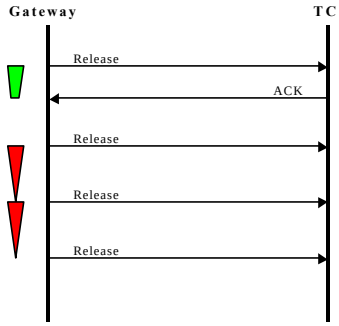
- Error Recovery Procedure -- Emergency



Network Technology Lab

Proposed System Architecture

- Error Recovery Procedure -- Release



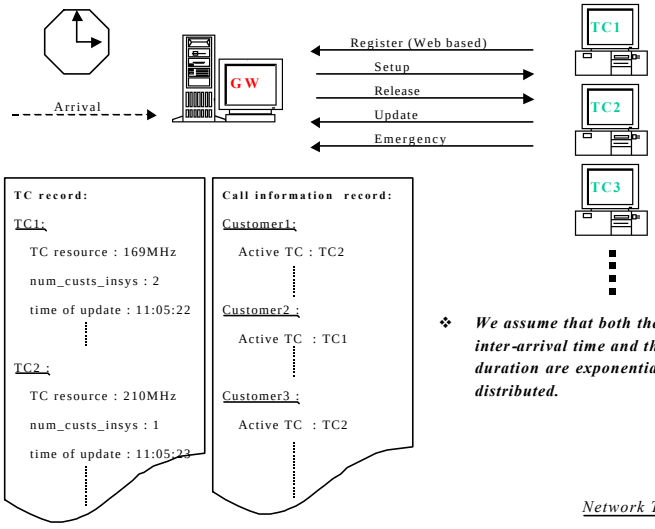
Proposed System Architecture

- Data Structure

- The gateway needs to maintain two tables:
  - The first one records the periodic update information of CPU utilization from each proxy-TC
    - CPU utilization
    - Update time
    - The number of serving users
  - The second one keeps the call information in the system
    - The active proxy-TC
    - Occupied port numbers
    - Call party information

Simulation Model and Results

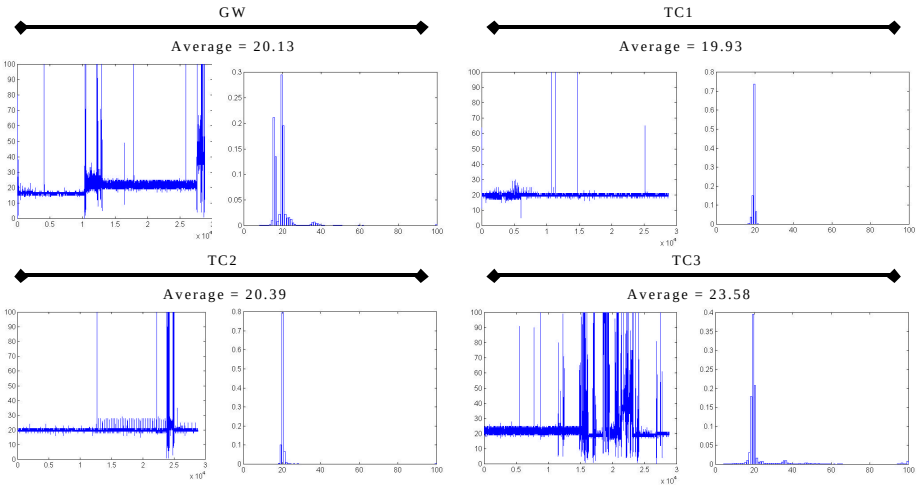
- The Simulation Model



❖ ***We assume that both the call inter-arrival time and the call duration are exponentially distributed.***

Simulation Model and Results

- CPU utilization records



Simulation Model and Results
- Proxy-TC selection schemes

- Definition:
  - Ri : available resource of i' th proxy-TC
  - Token(i) : selecting i' th proxy-TC
  - NR : necessary TC resource per call
  - N : the number of proxy-TCs
- First-Fit-Round-Robin (FFRR)

  $initial: i = 0$

  $while(R_i < NR) i = (i+1) \bmod (N+1)$

  $\{Token(i)\}$
- Maximum Available Resource

  $R_j = \max_i \{R_i\} \Rightarrow Token\ (j)$

Simulation Model and Results
- Performance Indices (1/2)

- Definitions of call events
  - Blocked call
    - When upon receipt of a new call, the individual CPU resources of the gateway and all registered proxy-TCs are less than 25MIPS.
  - Dropped call
    - When the serving proxy-TC is occasionally running out of resources for at least one second and no other proxy-TC can take over this call.

Simulation Model and Results
- Performance Indices (2/2)

- Definitions of probabilities
  - The new call blocking probability

$$P_b = \Pr[blocking] = \frac{\text{Number of blocked calls}}{\text{Total arrival calls}}$$

  - The active call dropping probability

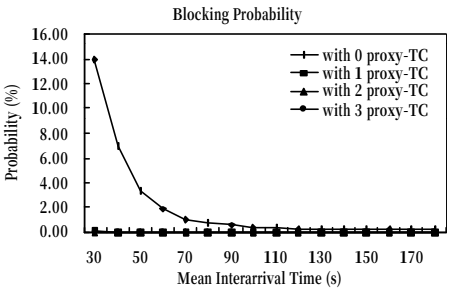
$$P_d = \Pr[droping] = \frac{\text{Number of dropped calls}}{\text{Total served calls}}$$

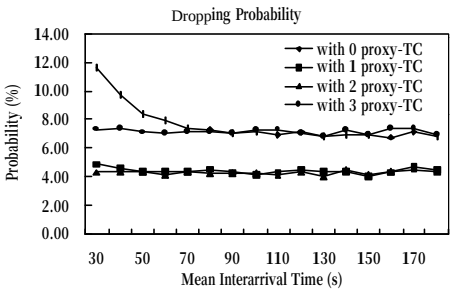
  - The handoff frequency

$$P_h = \Pr[Handoff] = \frac{\text{Number of handoff calls}}{\text{Total served calls}}$$

Simulation Model and Results
- The Simulation Results (1/5)

- *Case 1 :* The relation between the number of registered proxy-TCs and the probabilities of call blocking and call dropping, when *no handoff mechanism* is implemented.

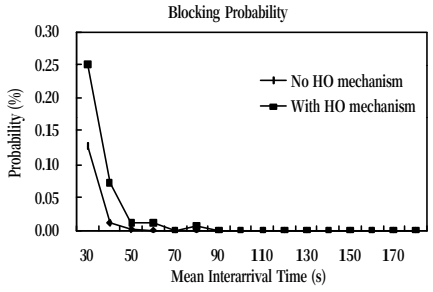


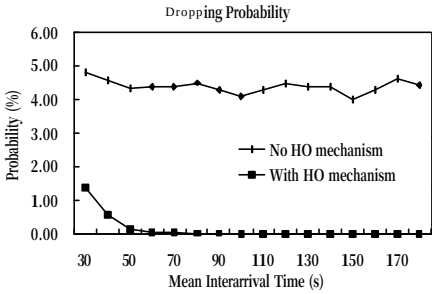


❖***The mean duration of each call is 3 minutes.***

Simulation Model and Results
- The Simulation Results (2/5)

- *Case 2* :The probabilities of call blocking and call dropping, when the gateway cooperates with only one proxy-TC and *handoff mechanism* is implemented.

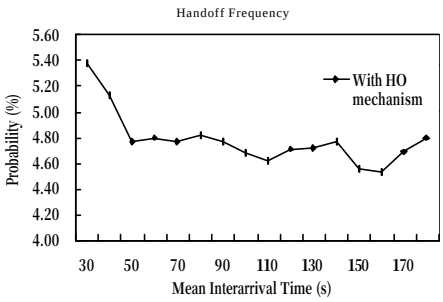




❖***The mean duration of each call is 3 minutes.***

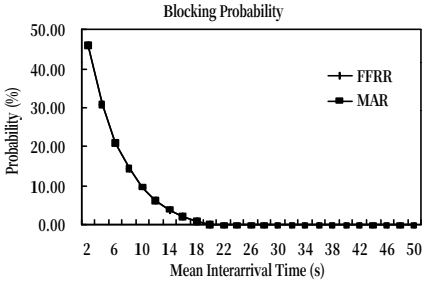
Simulation Model and Results
- The Simulation Results (3/5)

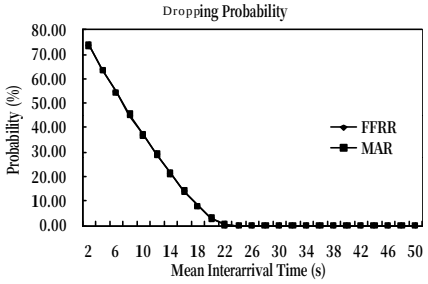
- *Case 2* : The handoff frequency, when the gateway cooperates with only one proxy-TC and *handoff mechanism* is implemented.



Simulation Model and Results
- The Simulation Results (4/5)

- *Case 3* : The performance comparison between the two call placement schemes (i.e., FFRR and MAR) under the condition that the gateway cooperate with 10 proxy-TCs and *handoff mechanism* is implemented.

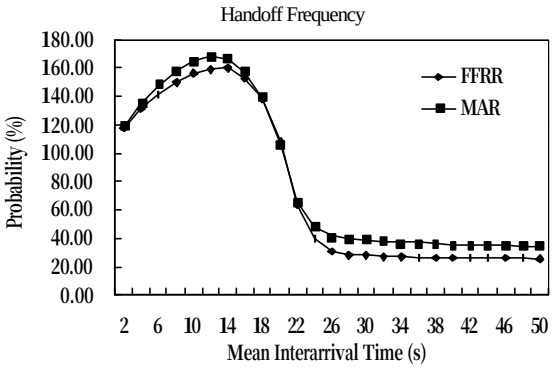




❖***The mean duration of each call is 15 minutes.***

Simulation Model and Results
- The Simulation Results (5/5)

- *Case 3* : The performance comparison between the two call placement schemes (i.e., FFRR and MAR) under the condition that the gateway cooperate with 10 proxy-TCs and *handoff mechanism* is implemented.



Conclusions

- We conclude that the proposed architecture achieves higher capacity and lower blocking probability with respect to the arrival calls, and the employment of the handoff mechanism leads to a drastic decrease in the active call dropping probability at the expense of a little increase in the new call blocking probability.
- Along this research direction, an interesting future work will be to find a method to predict the CPU utilization, as well as a good call placement approach to reduce the handoff frequency.