ABSTRACT

Title of Thesis: INTELLIGENT NETWORK BASED WIRELESS SERVICES FOR TELEMEDICINE

Degree candidate: Abhijeet S Bisain

Degree and year : Master of Science, 1998

Thesis directed by: Professor John S. Baras
Department of Systems Engineering

This thesis proposes an architecture for wireless services in telemedicine. The scenario visualized is of an ambulance carrying a trauma patient and sending medical data(video, ultrasound, ECG) to its corresponding hospital. It needs to know in advance the approximate bandwidth available to it after every handoff.

This thesis attempts to solve this problem with Intelligent Network signalling which aids fast implementation of new services. It is assumed that the ambulance uses a TDMA(GSM/PACS) based phone with multiple slot connection capability. New signalling procedures are suggested which attempt to provide this service with minimum delay and load.

Some slot allocation schemes implemented at the base stations are designed and evaluated. Buffer management schemes at the mobile to manage packets from various data streams are proposed and compared. All queuing systems are simulated in Opnet.

INTELLIGENT NETWORK BASED WIRELESS SERVICES FOR TELEMEDICINE

by

Abhijeet S Bisain

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
1998

Advisory Committee:

Professor John S. Baras, Chairman/Advisor
Professor Scott Corson
Professor Carlos Berenstein

DEDICATION

To my parents and Ritu for their and love and support.

ACKNOWLEDGMENTS

I wish to express my thanks to some of the many individuals who made this work possible. Foremost among these is my advisor, Professor John Baras, who gave me the opportunity to work on a very interesting and practical problem as part of the research for my thesis.

I am grateful to Dr. Carlos Berenstein and Dr. Scott Corson for kindly consenting to join the defense committee and review this thesis. I am also grateful to the Center for Satellite and Hybrid Communication Networks for the support of my research.

I wish to thank my friends and co-workers for their support and encouragement, especially Jagdeep and Varma.

Finally, I would like to thank my family for their love and support throughout my studies.

TABLE OF CONTENTS

List of Tables

List of Figures

INTELLIGENT NETWORK BASED WIRELESS SERVICES FOR TELEMEDICINE

Abhijeet S Bisain

August 3, 1998

This comment page is not part of the dissertation.

Typeset by LaTeX using the `dissertation` class by Pablo A. Straub, University of Maryland.

Chapter 1

Introduction

Telemedicine is the use of telecommunications for medical diagnosis and patient care. It involves the use of telecommunications technology for providing medical services to sites that are at a distance from the provider. This concept encompasses everything from the use of standard telephone service, through high speed, wide bandwidth transmission of digitized signals in conjunction with computers, fiber optics, satellites, and other sophisticated peripheral equipment and software. Telemedicine literally means, "medicine at a distance". The term was coined in the 1970s by Thomas Bird.

Telemedicine is a system that combines computer, video, and network communications and cost effective quality care. It includes diagnostic instruments designed to provide information for digital transmission and reproduction.

1.1 Telemedicine and its benefits

Telemedicine can be divided into three areas: aid to decision-making, remote sensing, and collaborative arrangements for the real-time management of patients at a distance. As an aid to decision-making, telemedicine includes areas such as remote expert systems that contribute to patient diagnosis or the use of online databases in the actual practice of medicine. Remote sensing consists of the transmission of patient information, such as electrocardiographic signals, X-rays, or patient records (history of illness etc.) from a remote site to a collaborator at a distant site. Collaborative arrangements consist of using technology to actually allow one practitioner to observe and discuss symptoms with another practitioner whose patients are far away. Two-way work stations which provide smooth digital motion pictures have been integral to the long distance, real-time treatment of patients [1].

One such application of telemedicine is when medics in an ambulance carrying a trauma patient collaborate with the doctors in their parent hospital. These medics transmit medical information (images, X-rays, ECG) to the doctors on the way to the hospital and receive operational instructions from them. This application falls in the mix of remote sensing and collaborative categories of telemedicine. We will concentrate on this aspect of telemedicine in this thesis.

Some of the potential benefits of telemedicine can be summarized as :

1. Improved access: Telemedicine can provide an improved access to health care in previously unserved or under-served geographical areas.

2. Reduced cost: The travel cost of the patients for specialty care, the travel cost of the health care professionals for continuing education or consulting, the personnel/equipment cost for not having to keep specialty care facility in rural hospitals, and other costs can either be eliminated or reduced.

3. Reduced isolation: Telemedicine provides a peer and specialist contact for patient consultations and continuing education. It has also been reported

that color, full-motion video is critical to the health professionals for simulating face-to-face communication between colleagues in consultations and between patients and physicians.

4. Improved quality of care: Telemedicine allows the consultation among the referring physician, the consulting physician, the patient, and the patient's family through interactive video with critical information of the patient available on-line. Also, the physicians or other personnel at remote locations can be educated during the consultations with specialty physicians and other experts, increasing their ability to treat other similar cases in the future. It helps the doctors to be better prepared for incoming patients.

1.2 Proposed Telemedicine Service

As we stated in the previous section, we will concentrate on the application of telemedicine, where an ambulance needs to collaborate with its parent hospital. In this thesis we propose a service to transmit high bit-rate medical data over the existing terrestrial wireless network from an ambulance (carrying a trauma patient) to the hospital. Studies have indicated that advance availability of the information about the patient's condition increases the chances of the treatment being successful by 20%. The doctors are better prepared when the patient arrives, with the right kind of instruments and medication, and they can better utilize the time in the ambulance by giving suggestions to the medics on-board.

The doctors need the patient's video, X-Rays, ECG and other information to make good decisions. In order to send this information, the ambulance needs to have a high-speed wireless communication link to the hospital. A study of expected data rates (obtained from Anaesthlab, UMBC) [2] of medical information is shown in the Table 1.1.

A	Motion Detection image	2kb×(8 to 30)frames/sec ≅ 16 to 60 kbps
B	High Resolution X-Ray	(static) 250kb (4in.×4in. image)
C	Chest X-Ray	(static) 1-2Mb (4×(B))
D	Good Quality Image	8kb (same motion requirement as (A))
E	ECG	1kbps - 10kbps
F	Ultrasound	250kbps
G	Audio	9.6kbps

Table 1.1: Typical Data Rates

A typical data connection in today's cellular network [3] infrastructure provides at most 19.2 kbps. A TDMA based standard called Personal Access Communication System (PACS) [4] promises a higher data rate capability of 28.8 kbps per TDMA slot. PACS also has the feature of multiple slot packet connections, making higher data rates (upto 214.6kbps) possible. Hence, PACS is chosen as the wireless network standard for our application.

In this thesis, we define a new service architecture for the mobile as shown in Figure 1.1. This service architecture makes high bit rate connections for the ambulance and informs it about the future availability of bandwidth. This is required because the bit-rate does not remain constant with time as the ambulance is handed off from one base station to another, with different base stations having varying capacities. The present network architecture lacks the signaling mechanism and the logic programs to perform this task.

As shown in Figure 1.1, we include Intelligent Network (IN) as an integral

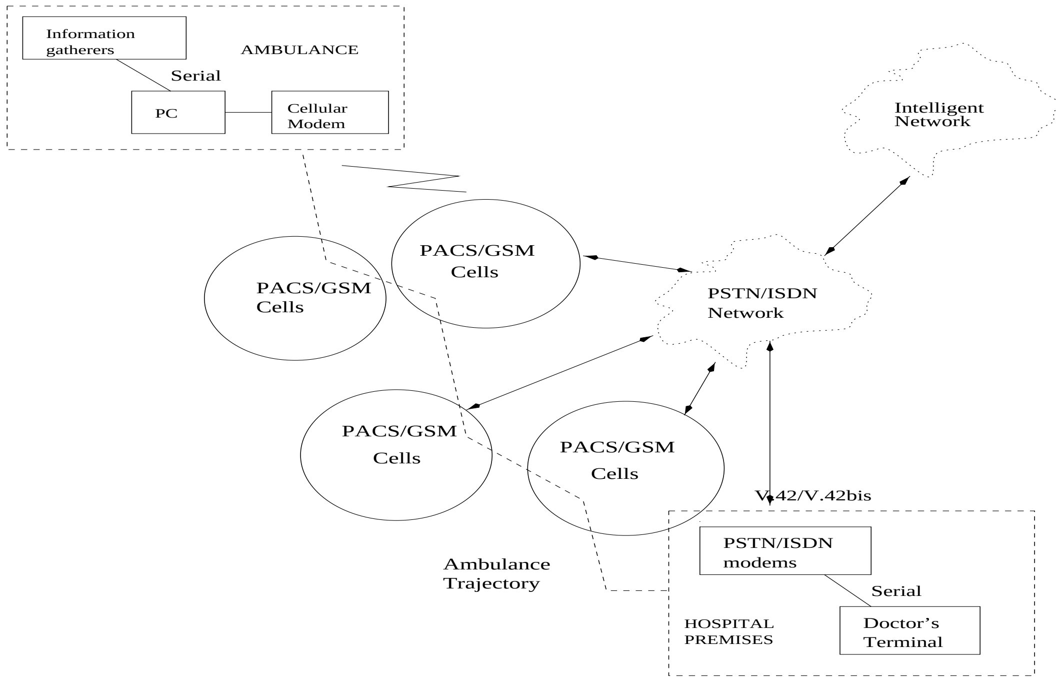

Figure 1.1: Service Architecture

element in our Service Architecture. We add features in IN such that it aids the ambulance in choosing optimal routes from the location of the patient to the hospital and in designing an optimal transmission schedule. Intelligent Networks (IN) [5] are a means of implementing new services over PSTN and ISDN networks. Under the present IN standard specifications, it is not possible to provide such a service; thus, new modules need to be added to support this service. This thesis outlines these modifications.

Other aspects of this service to an ambulance are also studied. The base stations in the PACS architecture need optimal slot allocation algorithms to maximize the slots allocated to the ambulance. We propose some slot allocation schemes and evaluate their performance. The ambulance requires mechanisms to optimally utilize its limited resources (memory space to queue packets and available time slots). We propose various queuing disciplines for the medical data to maximize the utility of information transmitted. Suggestions about the best slot allocation policies, queuing disciplines and network architecture for this service are made.

1.3 Organization of the Thesis

Chapter 2, in this thesis, introduces the Intelligent Network architecture and PACS air interface standard. Chapter 3 gives the details of the modifications to the present network architecture needed to provide this service to the ambulance. It includes the Service Independent Building(SIB) blocks, functional flow specifications, and Service Logic Programs needed to implement this service. In Chapter 4, several slot allocation algorithms for the base station and buffer management algorithms for the mobile are discussed. These are simulated in the Opnet simulation tool. We conclude in Chapter 5 with suggestions for future work.

Chapter 2

PACS and Intelligent Networks

In the previous chapter we highlighted the advantages of PACS and Intelligent Network standards. This chapter describes the relevant features of these standards needed to implement our service. PACS architecture and its benefits are introduced followed by a description of the CS-1 standard for IN.

2.1 PACS

2.1.1 Introduction

The Personal Access Communications System or PACS [4], is a low tier/low power radio system that has been standardized for operation in the 1850-1990 MHz licensed PCS band. It is being developed by Bellcore, HNS, NEC America, Parasonic, PCSI and Seimens. PACS provides an approach to PCS that is compatible with the local exchange telephone network and inter-operable with existing cellular systems. The system was designed to support mobile and fixed applications at low installation and operating costs while providing very high quality voice and data services. PACS capabilities include pedestrian and vehicular-speed mobility, data services, licensed and unlicensed systems, simplified network provisioning, maintenance and administration.

2.1.2 System Architecture

Figure 2.1 shows the PACS network architecture [6]. Three major elements in this architecture are the radio system, the Integrated Services Digital Network (ISDN)/Intelligent Network (IN) switch, and the IN Service Control Point (SCP). The radio system and switch communicate via ISDN protocol. The switch and the SCP communicate via AIN protocol. The AIN switch and the AIN SCP communicate with the public switched telephone network via Signalling System No. 7 (SS7) protocol.

The radio system consists of Subscriber Units (SU), Radio Ports (RP), and Radio Port Control Units (RPCU). The SUs communicate with the network through the RPs. The PACS air interface signal uses Time Division Multiple Access (TDMA) on the up-link [from SU to RP] and Time Division Multiplexed (TDM) on the down-link [from RP to SU]. Multiple RPs (e.g. 24-100) are connected to a RPCU through the Port (P) interface. The coverage area of an RP ranges from approximately a quarter of a mile to a mile. Transmission facility options for the P interface include E1, T1, high-speed digital subscriber line (HDSL) and digital subscriber line (DSL) technologies. The RPCU provides management and control functions between the RP and the local exchange network. RPCUs are connected to a standard class 5 switch through the Control (C) interface. The C interface uses an ISDN Basic Rate Interface (BRI). The RPCUs operate seamlessly with service providers subscriber database. This enables access to advanced intelligent network (AIN) services and features. The

RPs function largely as RF modems, depending on the centrally located RPCUs for most of the functionality. An advantage of locating most of the functionality in the RPCU is that service upgrades to support new data services or improved speech coders do not require visits to RP sites. The RPCU system automatically selects the best frequency for use by each RP, eliminating the need of detailed frequency management (Algorithm QSAFA).

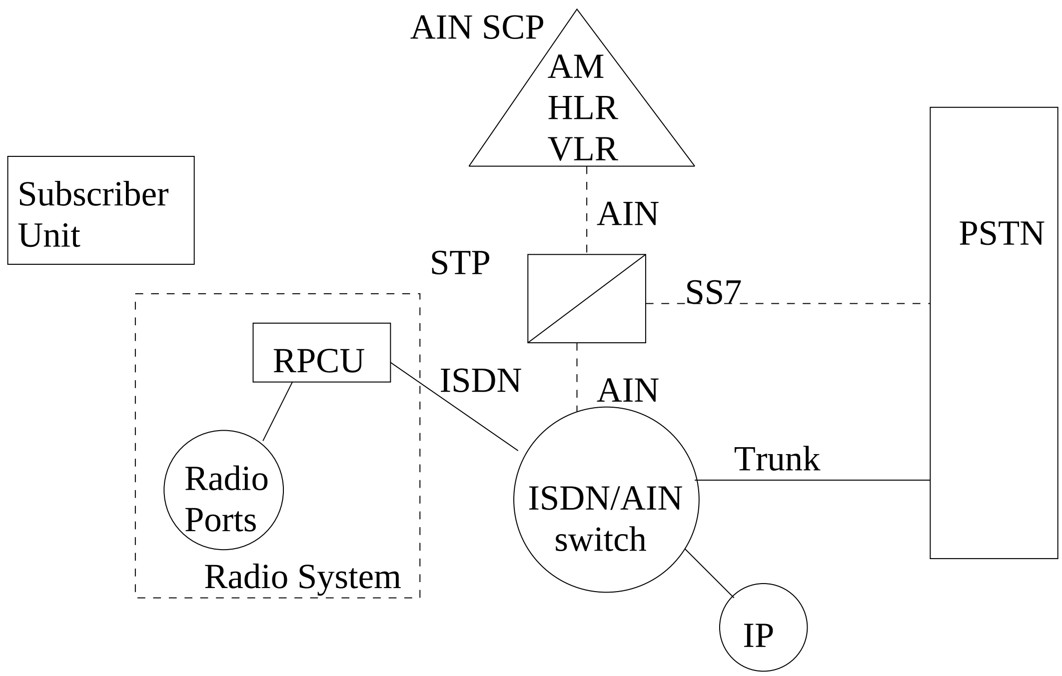

Figure 2.1: PACS Architecture

An Access Manager (AM) in the SCP supports multiple RPCUs with network-related tasks such as querying remote databases for visiting users, assisting in network call setup and delivery, and coordinating handoff between RPCUs.

The PACS air interface standard includes protocol specifications for an individual messaging service, a circuit mode data service, a packet mode data service as well as an interleaved speech/data service.

2.1.3 Benefits of PACS

Key features of PACS include:

1. Voice and data services comparable in quality, reliability and security with wire-line alternatives.

2. Very cost effective for serving high tele-traffic areas, small, inexpensive, line-powered radio ports for pole or wall mounting.

3. Low transmit power and efficient sleep mode requiring only small batteries to power portable subscriber units.

4. PACS is a low tier, low power technology. Antennae can be constructed quite inconspicuously, piggy-backing on existing structures. This avoids the high costs and delays attendant on permitting high towers.

5. PACS is based on Bellcore's WACS (Wireless Access Communications System), developed for wireless local loop replacement, and on Japan's Personal Handy-phone System (PHS). Several vendors are designing Subscriber Units for installation on subscriber premises.

6. Bellcore has designed PACS to support the full range of AIN services [6], including custom calling features, terminal and personal mobility, etc. As new AIN features are developed, the Bellcore standards defining this technology evolve to facilitate incorporation of the new services into PACS.

7. Control of frequency utilization lies at the RPCU, which is directly connected to the AM. The RPCU has abilities to mark channels for particular kind of users.

8. PACS supports priority and Emergency calling services. These calls can supersede ordinary calls and get access when they need.

2.2 Intelligent Networks

2.2.1 Introduction

The term Intelligent Network (IN) ([5], [7]) is used to describe an architectural concept applicable to all telecommunications networks. It provides a complete framework for the uniform creation, provision and management of advanced communication services. This is achieved by taking the data required for a particular service (e.g Free-phone, optimal routing,conference calling, call barring) and the service logic, outside the telephone switching network, and putting it into intelligent nodes.

Before IN, introduction of a new service required a change in the call handling software of every switch in the network. IN remedies this problem by taking the "intelligence" away from the switches, into the "intelligent modules". Each switch executes a Basic Call Process (BCP) for a call, and if it senses the requirement of a higher intelligence to support the call, it contacts the "intelligent agent". This agent contains the necessary processing tools and information (service logic) to understand the request made by the call and guide the switch on how to proceed with the call. These agents are called Service Control Points (SCP) and the switches which know SCPs exist are called Service Switching Points (SSP).

2.2.2 IN service processing model

The IN service processing model describes how any call gets processed in the network. The elements of this model are: the Basic Call Processes (BCP), the "hooks" that allow the BCP to interact with IN service logic, and IN service logic that can be "programmed" to implement new services ([8], [9], [10], [11]). The main principles for these elements are described below.

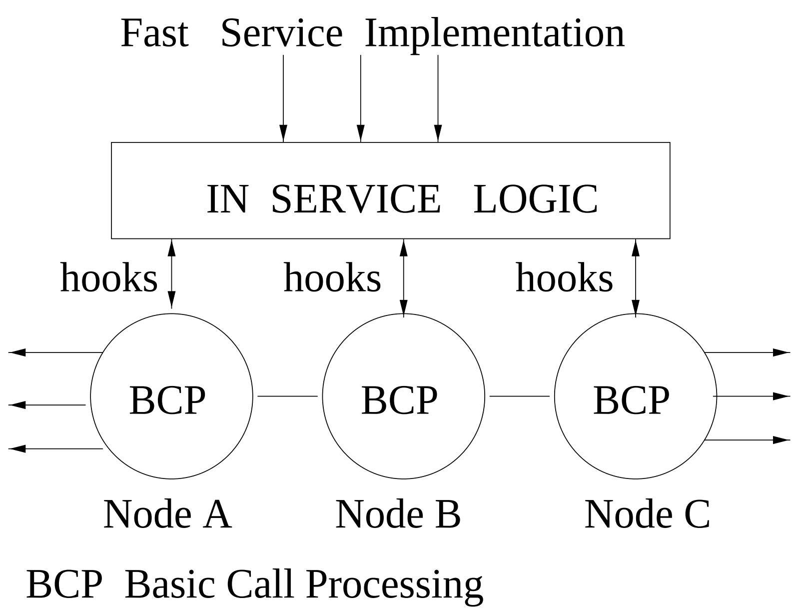

Figure 2.2: Typical IN service processing model

The basic BCP is available in all switches and is designed to support, with optimal performance, services that do not require special features. In order to achieve flexibility in service processing, the BCP is modularized into services-independent sub-processes such that these can be executed autonomously (without interference from the outside during execution).

"Hooks" are added to the BCP forming the links between the individual basic call sub-processes and the service logic. For this, the switches should continuously check the BCP for the occurrence of conditions on which an interaction session with IN service logic should be started. During an interaction session the BCP can be temporarily suspended.

The IN service logic uses a programmable software environment to execute the service logic. New supplementary services can be created by means of "programs" containing the logic for the desired service. It interacts with the BCP via the "hooks". It can decide to terminate an interaction session with the BCP. The basic call process will then resume its execution as specified by the IN service logic.

Thus, by changing the logic at the SCP and modifying network data, a new service that uses existing network capabilities can be readily implemented.

2.2.3 IN architectural concept

A key objective of IN is to provide service-independent functions/blocks (SIB) that can be used as "building blocks" to construct a variety of services.

The second objective is network implementation independent provision of services. This objective aims to isolate the services from the way the SIBs are actually implemented in various physical networks, thus providing services that are independent of underlying physical network infrastructure.

The IN Conceptual Model (INCM) (Figure 2.3) [12] is a framework for the design and description of the IN architecture which achieves the above objective. The INCM (Figure 2.3) consists of four "planes" where each plane represents a different abstract view of the capabilities provided by an IN-structured network. These views address service aspects, global functionality, distributed functionality and physical aspects of IN. IN Capability Set 1 (IN-CS1) is the first standardized stage of the Intelligent Network as an architectural concept for the creation [13] and provision of telecommunication services. The next few sections will discuss each plane shown in Figure 2.3.

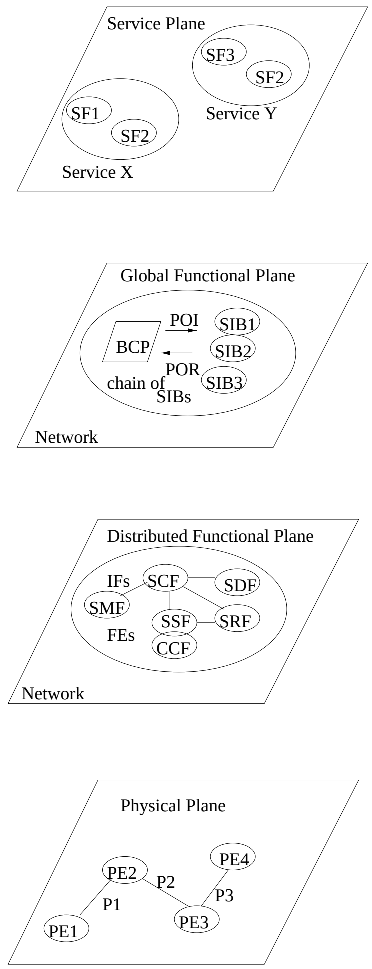

Figure 2.3: IN Conceptual Model

2.3 Service Plane

The service plane illustrates IN-supported services by means of a set of generic blocks called "Service Features". A service is a stand-alone commercial offering, characterized by one or more Service Features (core SF).The service plane represents an exclusively service-oriented view. This view contains no information regarding the implementation of the services in the network. All that is perceived is the network's service-related behavior as seen, for example, by a service user.

The presence of multiple services results in the interaction between the services and other supplementary services. Service interaction applies to all interactions of the service being described with other services already identified. An IN structured network handles multiple services for the same call. The service interaction is part of the specification of services.

2.4 Global Functional Plane

2.4.1 Global Functional Plane modeling

The Global Functional Plane (GFP) models network functionality from a global or network-wide point of view. The IN Structured Network is viewed as a single entity in the GFP. In this plane, Services and Services Features are redefined in terms of the broad network functions required to support them. These functions are neither Service nor Service-Feature specific and are referred to as Service Independent Building blocks (SIB).

The Global Functional Plane contains the following:

1. Basic Call Process (BCP): identifies the normal call process from which IN services are launched. This includes Points of Initiation (POI: a point in the call where the service is triggered), and Points of Return (POR: where the service logic returns and the BCP continues execution) which provide the interface from the BCP to the Service Logic Program. For a given

service, at least one POI is required; depending upon the logic required to support the service, multiple PORs may be defined.

2. SIBs: are standard reusable network-wide capabilities used to realize Services and Service Features

3. Service Logic Program (SLP): which describes how the SIBs are chained together, how they branch and the how the branches rejoin to describe Service Features. The SLP also describes interaction between the BCP and the SIB chains. Various network elements (SSP,SDP) can be contacted by the SCP during the execution of the SLP.

In order to chain SIBs together, knowledge of the connection pattern, decision options, and data required by SIBs must be available. SIBs, including BCP, are service independent and cannot contain knowledge of subsequent SIBs. Therefore, SLP is the only element in the GFP which is specifically service dependent.

2.4.2 Basic Call Process

The BCP is represented in terms of a high level finite state machine known as the Basic Call State Model (BCSM). The BCSM identifies points in basic call (PIC) and connection processing when IN service logic instances are permitted to interact with the basic call and connection control capabilities. PICs identify the activities required to complete one or more call/connection tasks of interest to IN service logic instances. Detection Points (DP) or the "hooks" indicate points in basic call and connection processing at which transfer of control (to the SLP) can occur (Figure 2.5). The transitions from one PIC to another indicate the normal flow of basic call.

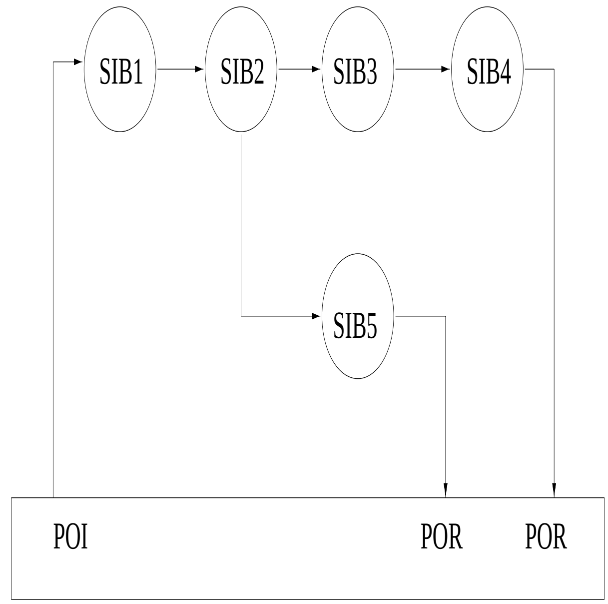

Figure 2.4: Modeling of Global Functional Plane

The BCSM also reflects the functional separation between the originating (Figure 2.6) and terminating portions of calls. Each of them is managed by a functionally separate BCP in the SSF/CCF. For IN-CS1, the following BCSM (Figure 2.6) is defined.

2.4.3 Points of Initiation and Points of Return in IN-CS1

The following set of POIs has been identified for CS1:

1. Call Originated.

2. Address Collected.

3. Address Analyzed.

4. Prepared to Complete Call.

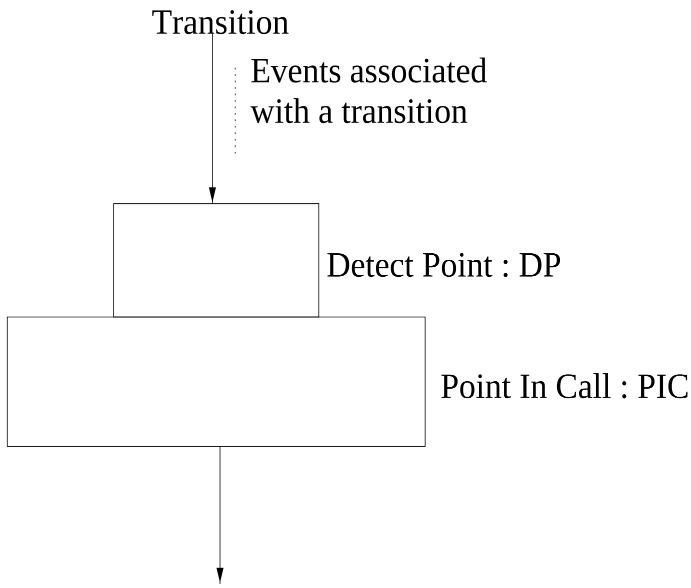

Figure 2.5: Key components of the BCSM

5. Busy.
6. No Answer.
7. Call Acceptance.
8. Active State.
9. End of Call.

The following set of PORs has been identified for CS1:

1. Continue with existing data: BCP should continue with no modification.
2. Proceed with new data: BCP should proceed call processing with only a data modification.
3. Handle as transit: BCP should treat the call as if it had just arrived.

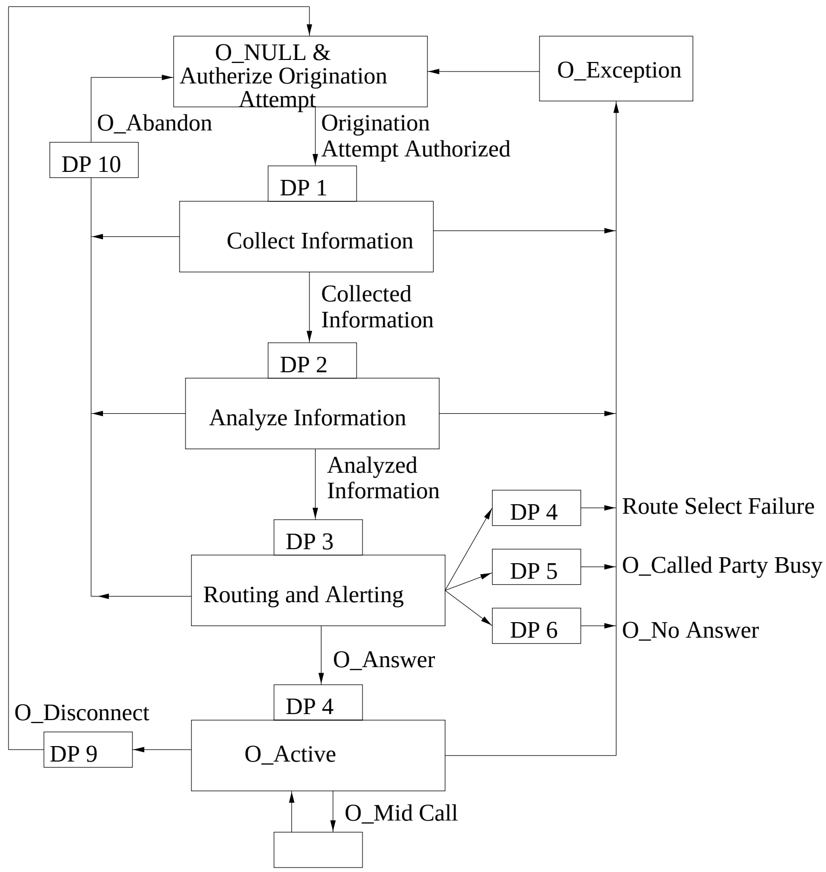

Figure 2.6: Originating BCSM for CS1

4. Clear Call: BCP should clear the call.

5. Enable call party handling: BCP should perform functions to enable call control for call parties.

6. Initiate Call: a call should be initiated independently or in the context of the call.

2.4.4 Service Independent Building Blocks

A SIB is a standard reusable network-wide capability residing in the Global Functional Plane used to create Service Features. SIBs are of a global nature and their detailed realization is not considered at this level but can be found in the Distributed Functional Plane (DFP) and the Physical Plane. Data required by each SIB is defined by Service Support Data (SSD) parameters, and Call Instance Data (CID) parameters, as shown in Figure 2.7.

CID defines dynamic parameters whose value will change with each call instance. They are used to specify subscriber specific details like calling or called line information. This data can be made available from the BCP (e.g. Calling Line Identification) generated by a SIB (e.g. a translated number), or entered by the subscriber (e.g. a PIN code).

SSD defines data parameters required by a SIB which is specific to the service feature description. When a SIB is included in the SLP of a service description, the SLP will specify the SSD values for the SIB.

The following set of SIBs of the IN-CS1 have been identified as required to support the list of targeted services:

1. **Algorithm**: apply mathematical algorithm to data to produce data result. It has been limited to an increment or decrement operation on the data.

2. **Charge**: determine special ways to charge for the call.

3. **Compare**: compare a value against a specified reference value.

4. **Distribution**: distribute calls to different logical ends based upon parameters.

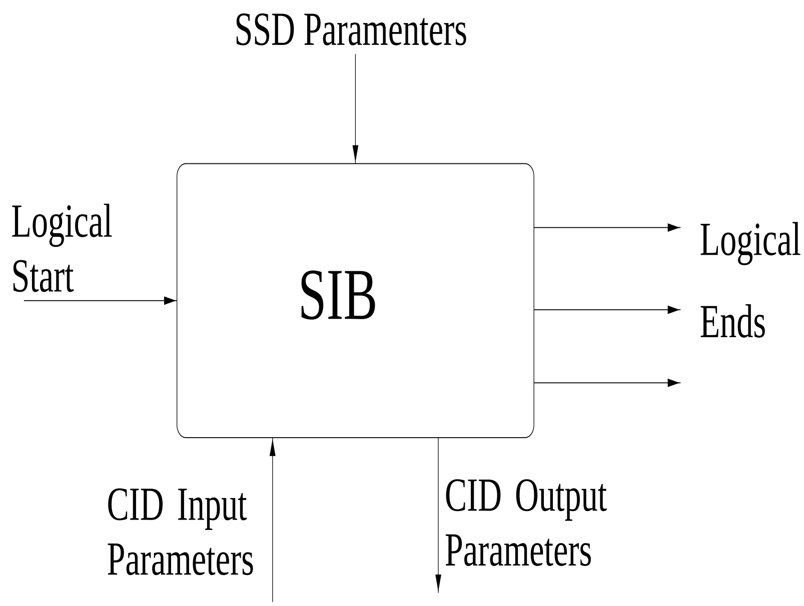

Figure 2.7: Graphic representation of SIB

5. **Limit**: limit the number of calls which use an IN provided service.

6. **Log Call Information**: log detailed information for each call into a file.

7. **Queue**: provide sequencing of IN calls to be completed to a called party.

8. **Screen**: determine if a supplied value exists in a list.

9. **Service Data Management**: replace, retrieve, modify user specific data.

10. **Status Notification**: inquiry about the status and/or status changes of network resources.

11. **Translate**: determine output information from input information based on a translation file.

12. **User Interaction**: information exchange between the network and a calling/called party.

13. **Verify**: syntactical consistency check of received information.

Figure 2.8 shows the representation of the screen SIB. Other SIB structures can be found in [12].

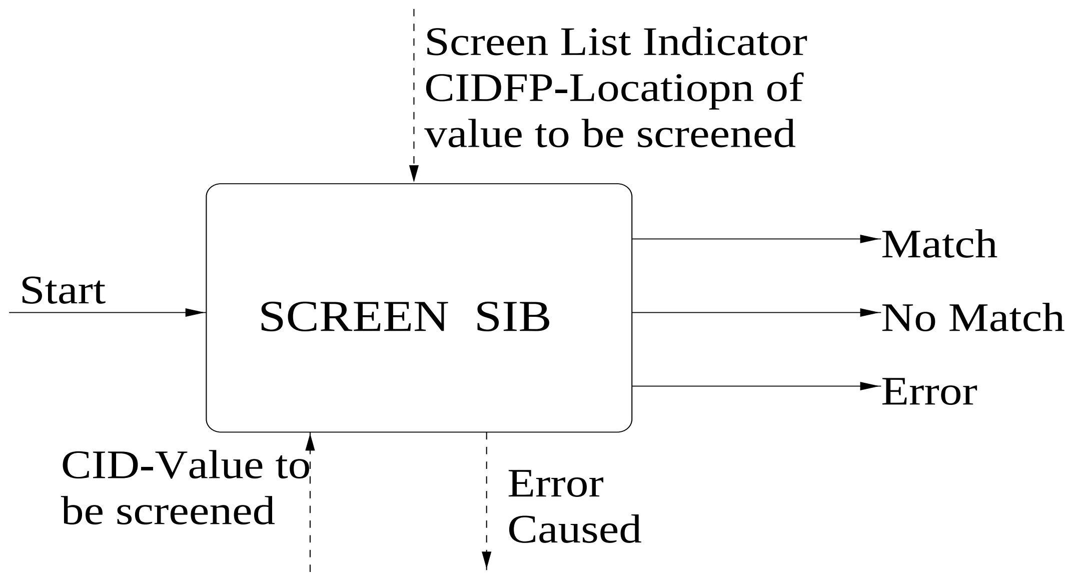

Figure 2.8: SIB example: Screen SIB

2.5 Distributed Functional Plane

2.5.1 Distributed Functional Plane Model

In this plane the network architecture is represented in terms of Functional Entities (FE). A FE (Figure 2.9) is a unique group of functions in a single location and a subset of the total set of functions required to provide a service. One or more functional entities can be located in the same Physical Entity (defined in the physical plane). Different functional entities contain different functions, and

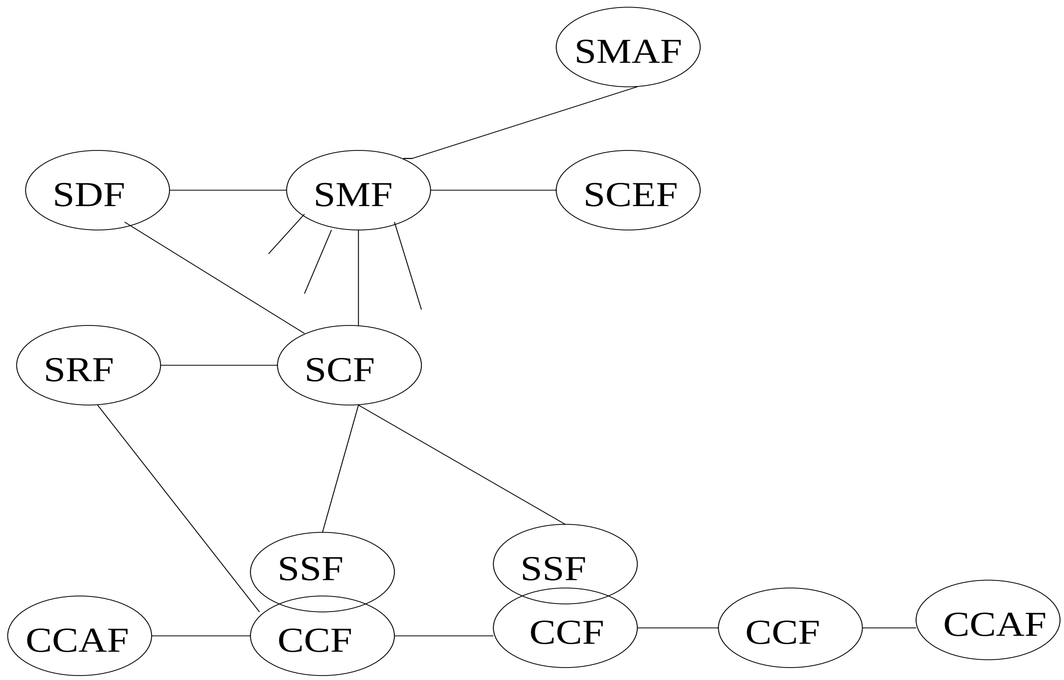

Figure 2.9: Functional Entities and Relations in the DFP

may also contain one or more of the same function. In addition, one functional entity cannot be split between two physical entities.

Each interaction between a communicating pair of functional entities in the model is defined by a set of Information Flows (IF).

2.5.2 DFP Functional Entities

The **Call Control Agent Function (CCAF)** provides access for users. It is the interface between user and network call control functions and receives indications relating to the call or service from the CCF and relays them to the user as required.

The **Call Control Function (CCF)** provides call/connection processing and control. It provides the capability to associate and relate CCAF functional entities that are involved in a particular call and/or connection instance. It contains trigger mechanisms to access IN functionality (e.g. passes events to the SSF).

The **Service Switching Function (SSF)** interacts between the CCF and a SCF. It modifies the call/connection processing functions (in the CCF) as required to process requests for IN provided service usage under the control of the SCF. The SSP sends traffic data including real time information to the SCP.

The **Service Control Function (SCF)** commands call control functions in the processing of IN provided or custom service requests. The SCF may interact with other functional entities to access additional logic or to obtain information (service or user data) required to process a service instance. The typical response time of an SCP for a service logic is less than 300ms.

The **Service Data Function (SDF)** contains customer and network data for access by the SCF in the execution of an IN service.

The **Specialized Resource Function (SRF)** provides the specialized resources required for the execution of IN provided services (e.g. digit receivers, announcements, conference bridges, etc.). It may contain logic and processing capability to receive/send and convert information received from users.

The **Service Creation Environment** allows services provided in IN to be defined, developed, tested and input to SMF. Output of this function would include service logic, service management logic, service data template and service trigger information.

The **Service Management Agent Function (SMAF)** provides a management interface between service managers and the SMF.

The **Service Management Function (SMF)** allows deployment and provision of IN provided services and allows the support of ongoing operation. It manages, updates and administers service related information in SRF, SSF and CCF.

2.6 Physical Plane

2.6.1 General requirements

The Physical Plane of the IN Conceptual Model identifies the different Physical Entities (PE) and the interfaces between these entities. One or more functional entities may be mapped onto the same physical entity.

The following selection of PEs has been defined to support IN.

1. Service Switching Point (SSP)
2. Service Control Point (SCP)
3. Service Data Point (SDP)
4. Intelligent Peripheral (IP)
5. Adjunct (AD)
6. Service Node (SN)
7. Service Switching and Control Point (SSCP)
8. Service Management Point (SMP)
9. Service Creation Environment Point (SCEP)
10. Service Management Access Point (SMAP)

Table 2.1 outlines a mapping from the FEs in the DFP to the PEs in the physical plane. The information flows between the FEs (in the DFP) are mapped to the TCAP messages between the PEs (in the Physical Plane). TCAP stands for Transition Capability Application Part.

	SCF	CCF/SSF	SDF	SRF	SMF	SCEF	SMAF
SSP	O	C	O	O	-	-	-
SCP	C	-	O	-	-	-	-
SDP	-	-	C	-	-	-	-
IP	-	O	-	C	-	-	-
AD	C	-	C	-	-	-	-
SN	C	C	C	C	-	-	-
SSCP	C	C	C	O	-	-	-
SMP	-	-	-	-	C	O	O
SCEP	-	-	-	-	-	C	-
SMAP	-	-	-	-	-	-	C

Table 2.1: Typical FE to PE mapping

C: Core O: Optional -: Not allowed

Chapter 3

Service Architecture

In the previous chapter, we described the PACS and IN standard chosen for the network. Now we model the network architecture on which the new service is proposed. We talk about our design of the new Service Independent Building (SIB) blocks, functional flow diagrams, and service logic programs needed to implement the service. Various implementation issues of the additional SIBs are discussed. The design of the SLP is critical for the proper functioning of the service. We propose two SLPs for two different services, differing in complexity and features provided. In designing the new SLPs, we emphasize particularly on the inputs and outputs of each SIB in the control sequence of the SLP. We identify the POIs and PORs for each SLP.

3.1 Network Architecture

For the development of the service proposed in this thesis, we first model the system we are working with. The system consists of the following entities :

1. The ambulance which has data capturing equipment connected to a PC through its ports. The PC software for compressing each data source to fixed size packets, and forwarding them to the wireless modem. The PC also has software to automatically re-dial when it looses the call, it has the facility to determine the location of the ambulance using GPS or other TDMA self location schemes and transmit this information to the network when requested.

2. PACS air interface is used between the wireless modem and the Radio Port Control Unit.

3. The Radio Port Control Unit is connected to the intelligent ISDN network through appropriate adaptors.

4. At the terminating end, the doctor's PC has ISDN equipment to receive calls and demultiplexing software to separate packets into ECG, video etc.

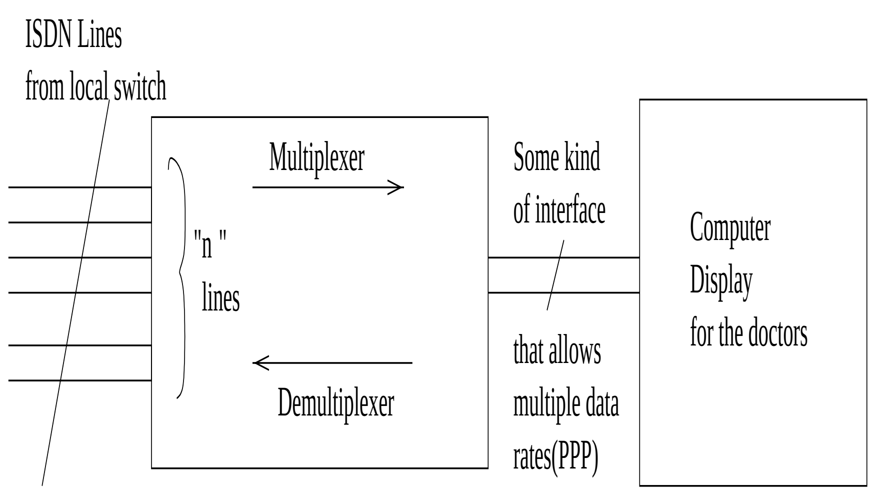

Figure 3.1: Hospital Hardware

In the network, the hospital has a set of ISDN lines coming into a lab as shown in Figure 3.1. It's software supports the BONDing protocol to setup a single connection using multiple lines and demultiplex the incoming data from multiple lines to the displays. The header of each packet has a data type field, and a packet number field. These are sufficient for the demultiplexing software to separate packets of different data types and sequence them before display. No end to end retransmission schemes are needed. Corrupt packets are dropped. If any data source in the mobile, generates a packet with size less than one time slot worth of bits, the remaining packet is stuffed with nonsense bits or checksums.

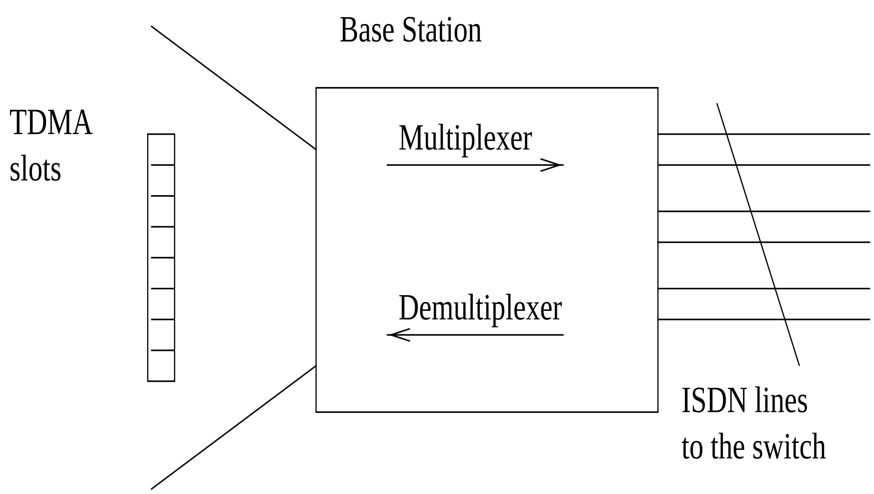

Figure 3.2: RPCU Hardware

The RPCUs (Figure 3.2) have the ability to setup a single connection on multiple ISDN channels and demultiplex data from one mobile (ambulance) to those channels according to some pre-specified algorithm. In effect, the base station receives data from the ambulance at multiples of 28kbps, which it transmits on multiple ISDN lines to the hospital. The number of ISDN lines chosen is dependent on the number of time slots allocated to the ambulance. For example, if the ambulance was allocated 4 slots, its consolidated bit rate would be 112kbps. So the RPCU will need 2B channels on the ISDN side to send the data of the ambulance to the hospital.

Since the connection of the ambulance is asymmetric, more bandwidth is required in the uplink than the downlink. This is exactly opposite to the requirements of some other applications which require more downlink and less uplink. This implies that the downlink channels can be shared with other users that need more downlink than uplink.

The Intelligent Network informs the ambulance about the bandwidth available to it after every handoff, helping it to formulate an optimal transmission schedule. With this information, the ambulance software can decide which data to send when, what coding scheme to use, what frame rate of video capture to choose etc.

It is envisioned, that this service can be implemented in two possible ways. In the first approach, the ambulance initiates the call setup from the patient's location. In this case, the SCP needs to determine all the SSPs on its route using its Geographical Information System (GIS). In the second approach the call is initiated when the ambulance leaves the hospital to pick up the patient. In this scenario, the base stations that the ambulance goes through, identify themselves to the SCP, send their present resource availability and reserve that resource. After the reaching the destination, the SCP sends this information to the ambulance. The latter is simpler to implement and guarantees higher bandwidth (advance reservation of capacity) to the ambulance, but is expensive because of the excess airtime. In this thesis the former approach will be dicussed in details.

3.2 New SIBS

3.2.1 SIB description

We described the Service Independent Building blocks in the previous chapter. The SCP executes these SIBs in a sequence given by the Service Logic Program to provide a service. In order to provide the service described in this thesis, we define additional SIBs not defined in CS-1. The new SIBs are described according to the following templates:

1. **Definition**: A detailed description of the purpose of the SIB.
2. **Operation**: A brief outline of the operation of the SIB.
3. **Input**: A list of input information required for the proper functioning of the SIB in terms of the SSD and CID specifications.
4. **Output**: A list of outputs (CID) generated by the SIB after its operation is complete and its output branches(SIBs).

CID stands for Call Instance Data and SSD stands for Service Specific data, as described in the previous chapter. Information is exchanged between SIBs, in terms of CID File Pointers (CIDFP).

We have attempted to design these new SIBs service independent.

3.2.2 SHORTEST PATH SIB

This SIB provides the shortest path between two nodes in a graph.

The inputs to this SIB are: the CIDFP to the source node, to the sink (destination) node and to the address of the graph data structure.

The output is a CIDFP to an ordered list of edges on the shortest path. The output is an Error if the path could not be found.

3.2.3 UNIQUE LIST SIB

This SIB takes a matrix as an input and outputs a new matrix in which no two consecutive rows are equal; keeping one copy of the repeated rows.

The input to this SIB is a CIDFP, pointing to the matrix and the row and column lengths of the matrix.

The output is a CIDFP, pointing to the result matrix and its dimensions.

3.2.4 MULTIPLE SCP_SSP SESSION SIB

This SIB initiates multiple sessions between one SCP and several SSPs. The sessions are parallel or serial depending on the implementation.

The input is a pointer to list of SSP ids that need to be contacted and a pointer to the TCAP message.

When the SIB is multithreaded, it spawns multiple threads depending on the number of SSPs supplied. In case of serial sessions, no multithreading is required, instead a loop is executed as many times as the number of SSPs. During each loop, the current SSP is contacted, its response is obtained, the session is terminated, and the current SSP id is set to the next SSP id in the input list.

The outputs of this SIB is a list of CIDFPs pointing to the data obtained from the various SSPs, or Error if any of the SSPs did not respond within the specified time interval.

3.2.5 ADJUST EDGE-WEIGHT SIB

This SIB modifies the weights of the edges in a graph depending on the data provided to it.

The input to this SIB is a pointer to a list of edges, a pointer to an $n \times m$ matrix containing 'm' attributes of 'n' edges, and a pointer to a function for calculating the weight of an edge depending on the attributes of that edge.

The ouput is a list of edges with the updated weights.

3.3 Information Elements

As described in Chapter 2, Information Elements are the packets (TCAP) exchanged between two functional entities. To ensure proper functioning of this service, some new information elements are required.

1. Between the RPCU (SSP) and an SCP, an IE about the emergency call (which is a trigger query to distinguish this call from other regular phone calls).

2. From the SCP to an SSP to invoke and inquire about their bandwidth availabilities. The Register (P-INFO) message format can be used to do this.

3. From all SSPs that receive the above message, to notify the SCP, about their resource availability (Return Result Monitor Resource message in TCAP).

4. From SCP to an SSP to release reserved resources.

3.4 Functional flow diagram

The functional flow diagram of the service is depicted in Figure 3.3. It shows the different components of the networks and the information flows between them.

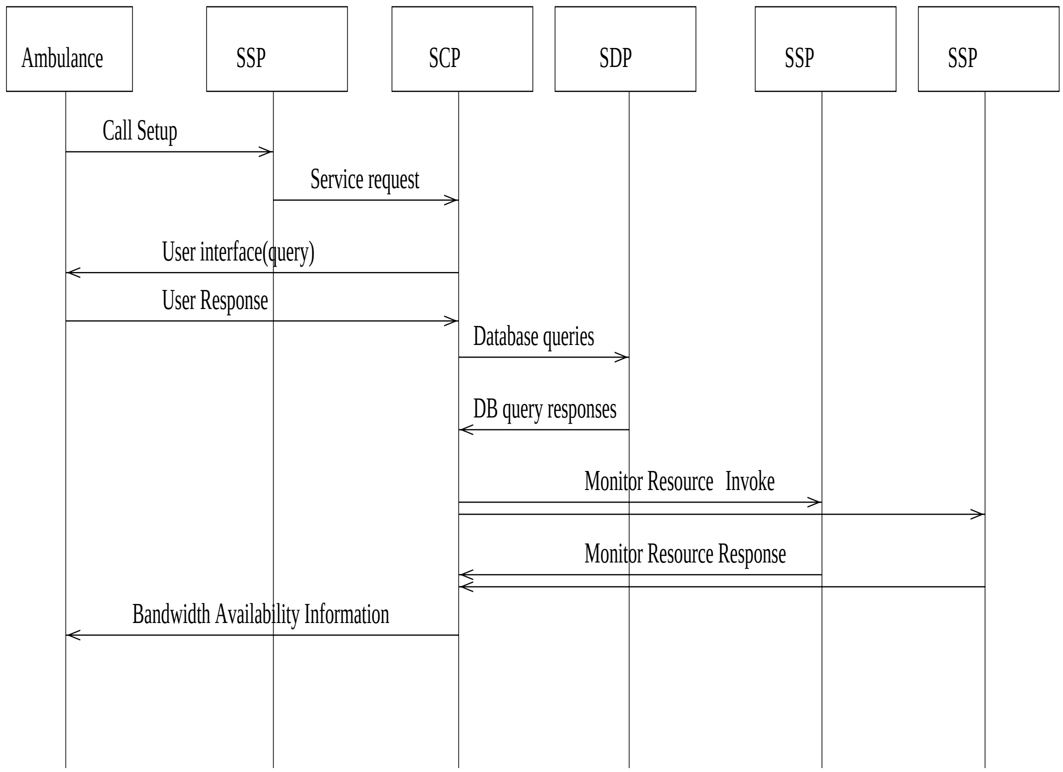

Figure 3.3: Service Information Flow Diagram

The following messages are exchanged between the physical entities.

1. The SSP, when realizes that the call is a special one, sends a message to the SCP.

2. After receiving this message from the SSP, the SCP (User Interface SIB) queries the user for data.

3. After receiving this message from the SSP, the SCP sends database queries to the SDP to obtain data required by its SIBs.

4. Using the SSP ids provided by the SDP, the SCP sends a message to all the RPCUs, requesting them to provide the maximum bandwidth available and reserve that bandwidth.

5. The RPCUs reply back with a Return Result message containing the bandwidth data. The RPCUs have the capability to mark availible traffic channels as busy and use them for emergency calls. If further slots become availible before the ambulance enters it region, the RPCU can mark those slots busy too.

6. The SCP (User Interface SIB) sends this information to the mobile.

3.5 The Service Logic

When the ambulance makes a call, the IN switch (SSP) sends a query to the IN SCP. Based on the trigger type and the parameters of the message the query is processed by the SLP in the IN SCP. The SSP detects a special call after the AnalyzeInfo PIC and before the route is selected and triggers the service logic (Figure 3.4). AIN0.1 has trigger points for N11 type calls. In our Service Logic, the following SIBs are used in the service logic:

1. The **Screen** SIB is used to authenticate the user. The inputs are :

   (a) Screen List Indicator (SSD):List of users (telephone numbers) subscribing to this service.

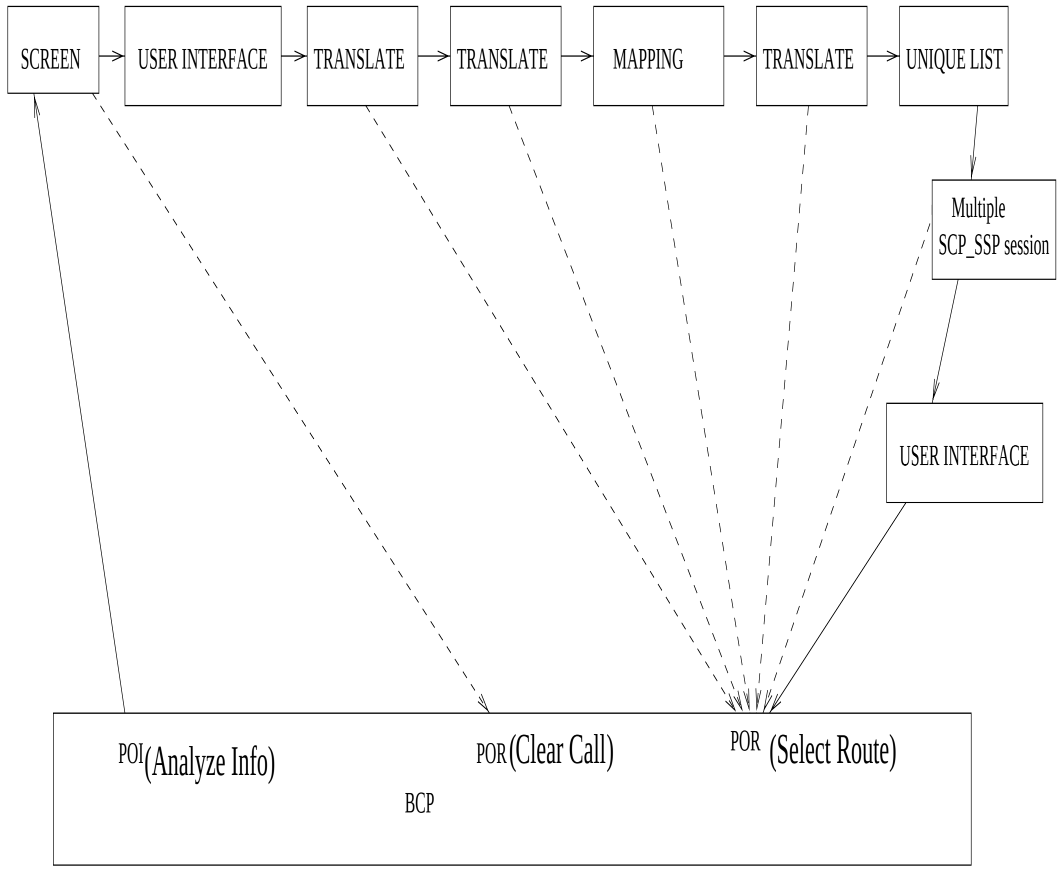

Figure 3.4: The Service Logic Program

(b) CIDPF:Location of the value (telephone number of the ambulance) to be screened.

The output is a "Match" if the user is found and the control goes to the next SIB decided by the SLP. It is a "No Match" if the user has not subsribed to the service; the control goes back to the BCP in the Clear Call PIC and the call is cleared.

2. The **User Interface** SIB is used to obtain the ambulance's longitude and latitude and the hospital's telephone number. The inputs to this SIB are:

(a) SSD-Activity: Play announcement and collect data.

(b) SSD- Announcement Indicator.

(c) CIDPF-Location of announcement parameter: “ Give latitute and logitude of your location”.

(d) CIDPF-Location of the collected information: CIDPF_1.

The output of this SIB is always a “Success”.

3. The **Translate** SIB is used to obtain the crossroads closest to the Longitude and Latitude provided by the ambulance. The inputs to this SIB are:

   (a) SSD-Translate File Indicator: Longitude,Latitude to crossroads.

   (b) CIDPF-Location of the value to be translated: CIDPF_1.

   The output is CIDPF_2 pointing to the name of the crossroad if the translation is a success, else an error and the control returns to the BCP to continue the call (Select Route).

4. The **Translate** SIB is used to obtain the crossroads closest to the hospital phone number provided by the ambulance. The inputs to this SIB are:

   (a) SSD-Translate File Indicator: Hospital phone numbers to crossroads list.

   (b) CIDPF-Location of the value to be translated: Hospital phone number.

   The output is CIDPF_3 pointing to the name of the crossroad if the translation is a success, else an error and the control returns to the BCP to continue the call (Select Route).

5. The **Shortest Path** SIB is used to find the shortest route between these crossroads. The city is represented in the form of a "Graph" [14] with the street intersections as the "Nodes" in the Graph, and the streets connecting these intersections as the "Edges" in the Graph. The edges are weighted by the length of the streets they represent. The shortest distance is calculated by applying Djikstra's algorithm [14] .

   Inputs to this SIB are:

   (a) SSD-Database: Graph which maps the city.

   (b) CIDPF-Location of the source and destination points (CIDFP_2 and CIDFP_3).

   The output is a CIDFP_4 pointing to a list of edges on the shortest path. If there is an error, the control returns to the BCP to continue the call (Select Route).

6. The **Translate** SIB is used to translate the edges obtained from the Shortest Path SIB to cell ids and the RPCU ids in the PACS network that these streets come under. The inputs to this SIB are:

   (a) SSD-Translation File Indicator: Streets (edges) to cell ids and RPCU ids.

   (b) CIDPF-Location of the values to be translated:List of edges (CIDPF_4).

   The output ("Success") is CIDFP_5 pointing to an $n \times 2$ array with RP ids and RPCU ids covering the path of the ambulance. Consecutive rows in this list correspond to a handoff between cells. It is possible that two consecutive edges map to same cells. This corresponds to one RP covering different streets. This situation is possible in reality, but we must remove such entries from our list because they do not represent a handoff. If an error, the control returns to the BCP to continue the call (Select Route).

7. The **Unique List** SIB is used to form a modified list from the list provided by the previous Translate SIB. The input to this SIB is CIDFP_5 and the output is a unique-elements list CIDFP_6.

8. The **Multiple SCP_SSP Session** SIB contacts the RPCUs in this unique list to get the bandwidth availibility in the specified RPs in its control. The message is the Monitor Resource message to inquire about bandwidth availability at the RPCU. The inputs to this SIB are:

   (a) CIDPF:the message to be sent to the RPCU.

   (b) CIDPF:The list of the RPCUs (with the cell ids) to be contacted (CIDFP_6).

   The responses obtained from the SSPs are pointed to by the output pointer: CIDFP_7. An AIN0.1 SCP has the capabilities to monitor resources. This enables the SCP to request the switching system to monitor the busy/idle status of a particular line and reflect changes.

9. The **User Interface** SIB conveys this message (CIDFP_7) to the ambulance. The SCP sends a "send to resource" message to the switch system to have the IP play an announcement. The SSP connects the ambulance to the IP via SSP-IP ISDN interface. The inputs to this SIB are:

(a) SSD-Activity (play announcement).

(b) SSD-Announcement Indicator.

(c) CIDPF-Location of announcement parameter. (CIDPF_7)

(d) CIDPF-Location of collected information (NULL).

The control returns to the BCP at the Select Route PIC.

Now, that the O_SSP and the T_SSP know how many channels are needed for connection, they setup a standard ISDN call on all these channels.

3.6 Best-Path Service

In this section, we describe another service, that is more complex than the previous but provides a better result. This service finds the best path for an ambulance, that is, a path with the best combination of distance and bandwidth. The Information Flow for this service is similar to Figure 3.3.

3.6.1 Service Logic Program

The following are the SIBs executed in the order given, to perform the logic for the service.

1. **Screen**: This SIB verify that the calling party (the ambulance) subscribes to the service.

2. **User Interface**: This SIB connects to the user, and obtains the logtitude and latitude of its location and the phone number of the hospital.

3. **Translate**: This SIB translates the hospital phone number to its longitude and latitude information, using approriate tables. The table are supplied by the service logic.

4. **Service Data Management**: This SIB performs the Retrieve operation twice to retrieve the graph of the part of the city lying in the rectangle enclosed by the longitude-latitude of the ambulance and the longitude-latitude of the hospital. The inputs, are the CIDFPs to the longitude-latitude information and the database containing the city's map. The output is a list of the nodes and the edges in the graph.

5. **Translate**: This SIB translates the names of the edges obtained from "4" to the RPCU and cell ids covering those edges. The inputs are the CIDFPs pointing to the list of edges obtained from "4" and the CIDFP pointing to the table mapping the edges to the cel and RPCU ids. The output is a list of cell and RPCU ids (CIDFP_5).

6. **Multiple SCP_SSP Session**: This SIB contacts all the RPCUs pointed to by the CIDFP_5. The input is CIDFP_5 and a pointer to the message to be sent to the RPCUs. The output (CIDFP_6) is a pointer to the results obtained from the RPCUs.

7. **Adjust Edge-weight**: This SIB is a HLSIB, which uses the Algorithm SIB multiple times to adjust the weight of the edges. We use this SIB to manipulate the weights of the edges in the graph, so that , the weight is a function of the length (time) required to traverse an edge, and the bandwidth available on the edge. This function is a Cobb-Douglas production function of the form :

$$Weight = a_0 Distance^{a_1} Bandwidth^{a_2}$$

Where, a_1, and a_2 depend on the importance of Distance and Bandwidth. The medic can decide the Marginal Rate of Substitution (MRS) for the system and using the MRS, the relative values of a_1 and a_2 can be determined. The exact values are not required because the Weights will be used for comparison between edges.

The Bandwidth value in the above equation, is the average bandwidth available on an edge, because, in real cellular systems that, an edge may be served by multiple RPs and each RP has a different amount of bandwidth available. If any edge has no available bandwidth, on even a small portion of its total stretch, the edge must be given zero Weight. Because, this means that the ambulance will be dropped and it will have to redial. Redials are extremely costly, as the new call has to go through the same process again, and must be avoided.

The output of this SIB is CIDFP_7, which points to the updated graph.

8. **Translate**: This SIB translates the longitude-latitude information of the ambulance's position to the closest node. The output CIDFP_8 points to the result node.

9. **Translate**: This SIB translates the longitude-latitude information of the hospital to the closest node. The output CIDFP_9 points to the result node.

10. **Shortest Path**: This SIB finds the shortest path between the nodes pointed to by CIDFP_8 and CIDFP_9 in the graph pointed to by CIDFP_7. The output CIDFP_10 is a list of nodes and edges defining the path from ambulance's present location and the hospital.

11. **Translate**: This SIB translates the edges in the path (pointed to by CIDFP_10), to the cell-ids and RPCU ids serving those edges. The output (CIDFP_11) is a list of cell ids and RPCU ids.

12. **Multiple SCP_SSP Session**: This SIB sends a "reservation release" message to the RPCUs in the list CIDFP_5 and not in the list CIDFP_11. This will free the extra reservation done in step 6. Another way to achieve this goal

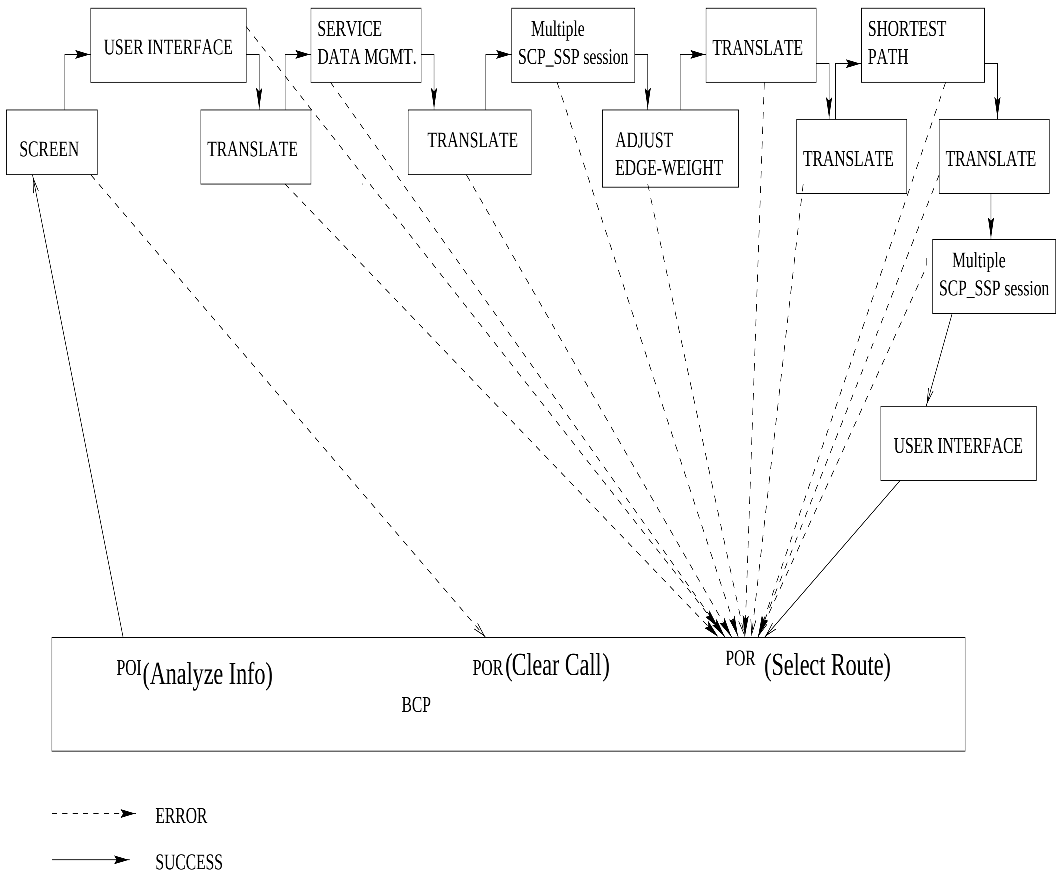

Figure 3.5: The Service Logic Program

is to tell the RPCUs in CIDFP_11 that their services will be required and the other RPCUs free their resources after a timeout. The downside of this approach is that, if the connection between an RPCU (in CIDFP_11) and the SCP goes down the RPCU will not realize that its services are needed and will free its resource after a timeout.

13. **User Interface**: This SIB sends the list CIDFP_11 to the ambulance.

3.7 Design Analysis

We have given a comprehensive description of the service architecture in the last section. In this section, we analyze the design critically, and explain why the new elements mentioned, are required. We also discuss other applications of these elements and why they should be included in the standards.

The importance of the bandwidth information has been discussed before. In the PACS standards, there is no provision for inter-RPCU signaling. An Access Manager controls some functions in the RPCU but lacks access to RPCUs beyond its domain. This is one very important reason why we included IN in our architecture. The signalling of IN makes possible such communication between the SCP and RPCUs as required by the service.

In the current IN standards, there are only a few limited SIBs, and the standards await final decisions. The standards define new modeling constructs called, High-Level SIBs, which are "entities composed of SIB operations and/or other SIBs". These are like macros, and consist of one or more of the 16 basic SIBs. The SIBs we proposed can also be viewed as HLSIBs and be constructed by using the 16 basic SIBs. This construction, however, was found to be very complex and a separate definition was deemed essential.

The information elements that our service architecture used, are not currently available in their exact form. In the present standards, the SCP can request an SSP to monitor a resource, and send information about the resource. But these resources are usually, the telephone lines, or the peripheral equipment. For our "wireless network" application, the resources are the frequency channels and the TDM slots on these channels. So, a message format to request information about a specific channel and a slot must be devised. This message has other frequency planning applications too.

The J-STD-014 standard for PACS, supports multislot priority calls. These calls receive access even if all the channels are currently being used. The RPCU can mark slots for priority users and can set information in the downlink broadcast channel for available services on each channel. This is crucial for our application, where we require the RPCUs to reserve channels for the ambulance. If all the traffic channels are busy, then a priority or emergency caller can communicate its request to the RPCU, through the uplink of the Systems Broadcast Channel and gain access to a traffic channel.

Chapter 4

Evaluating Slot Allocation Schemes (for RPCU) and Queue Management Schemes (for Ambulance)

In the earlier chapters, we developed the network architecture to provide the telemedicine service over the terrestrial wireless loop. This approach can be easily modified to serve other similar applications, where high bitrate data needs to be exchanged between distant locations. By incorporating Intelligent Networks, we have achieved the signaling capability to achieve the information flow required by the service implementation.

The introduction of the multislot capability at the RPCU raises a need for optimal slot allocation schemes. There are two kinds of customers at the RPCU, the single slot users, namely the normal telephone calls, and the multislot users, namely, the ambulance and other high data rate users. In this chapter we evaluate some slot allocation algorithms to optimize the slot allotment to the ambulance.

Presence of multiple sources of medical data at the ambulance and limitations in its buffering capacity necessitates design of optimal queue management algorithms [15]. We propose and evaluate some queue management schemes for the ambulance. The simulation results for these algorithms are obtained using the Opnet simulation tool.

4.1 Slot Allocation at the RPCU

For the analysis of the slot allocation problem, we assume that the RPCU (Figure 4.1) contains **n** frequency channels, each containing **c** TDM slots. There are be two types of users arriving in the system. The first type is a single slot user, which is the common cellular phone call. The second type is a multislot user which needs as many slots($\leq$ **c**) as available. In this case, the ambulance trying to send medical data, is a multislot user.

We study and simulate different slot allocation algorithms. In these simulations we assumed 8 channels each containing 8 TDMA slots. We chose these numbers because the PACS air interface standard specifies the use of 8 slots per channel, and each RPCU allocates approximately 8 channels per RP. Three statistics of each algorithm are recorded and compared. The three statistics chosen are:

1. The average number of slots allocated to the multislot users .

2. The blocking probability for single slot (SS) users.

3. The blocking probability for multislot (MS) users.

Our state diagram (as implemented in Opnet) of this queuing system is shown in Figure 4.2. It consists of five states: **init**, which initializes the queues

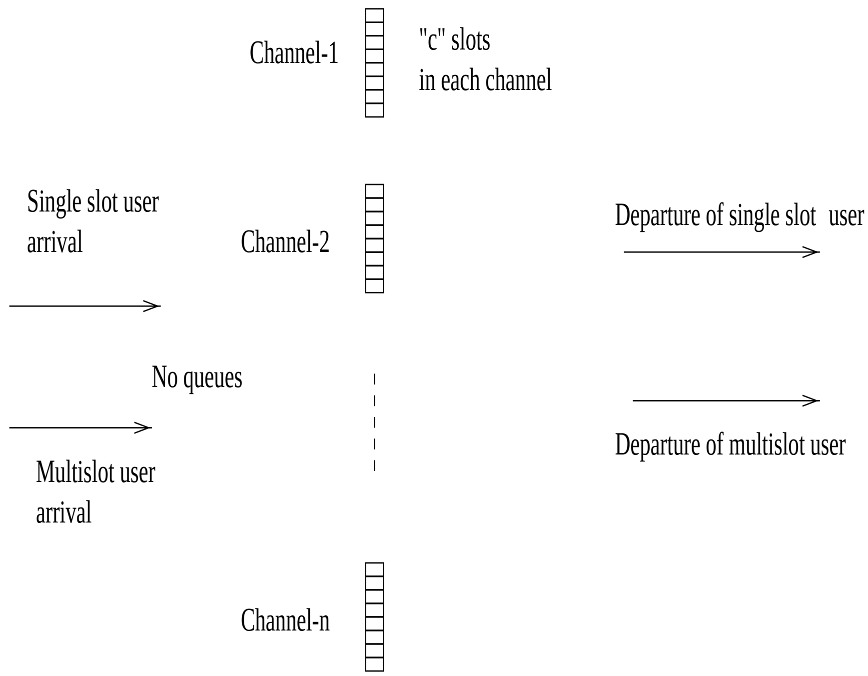

Figure 4.1: RPCU Queuing System

and statistic vectors; **arrival**, which allocates slots to arriving calls decided by the specific algorithm; **svc_st**, which initiates the service of the call; **svc_cpl**, which terminates a call; and **idle**, where the system waits for an event to occur. The arrows in the diagram defines the transition relation between the states. The transitions are : **ARRIVAL**, which indicates the arrival of a customer;**SVC_COMPLETION**, which indicates the end of a particular call; **insert_ok**, which indicates that an arriving call was successfully allocated channel(s); and **default**, which indicates that no event took place.

The arrival processes are assumed to be Poisson and service rate (Table 4.1) are assumed to be Exponential. The queue size is equal to the number of servers. So, this is an M/M/m/m system (where m = $\mathbf{n} \times \mathbf{c}$), except that the service policy is different from the conventional FCFS (First Come First Serve). Hypothetical

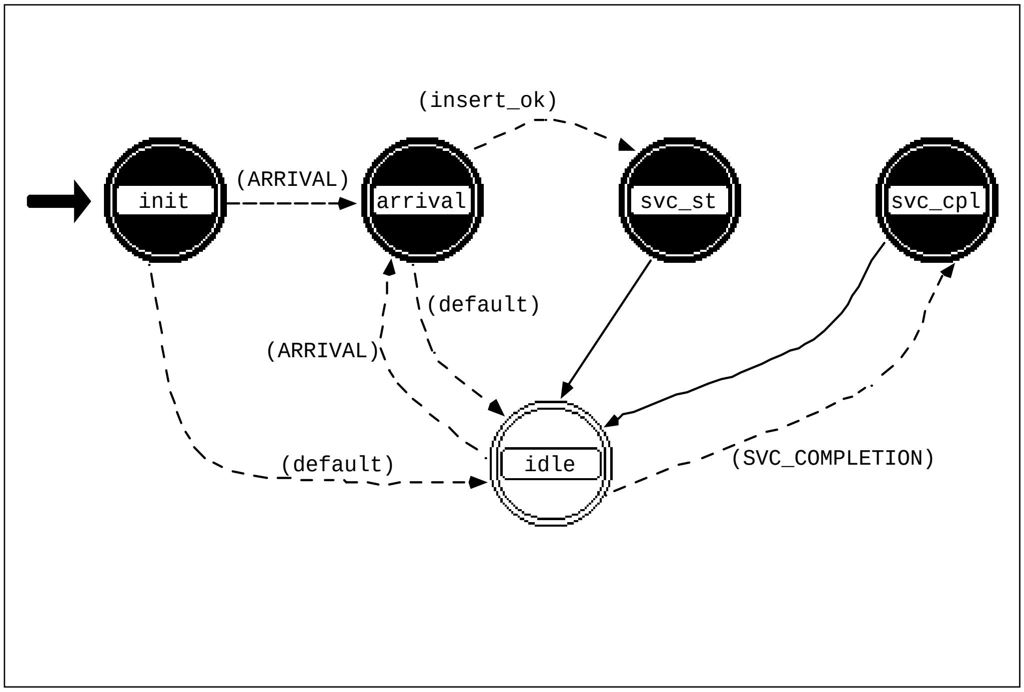

Figure 4.2: State diagram for the base station queue

arrival rates are chosen to differentiate the performance of the algorithms and are close to the real situation numbers.

4.1.1 Single slot users to busiest channel

In the simplest scheme, the single slot user is allocated the first available time slot and the multislot user is allocated all the free slots on the first available frequency channel. This scheme can be made more efficient by allocating the single slot user, a slot from the busiest channel, i.e., the channel that has maximum number of busy slots and atleast one idle slot. The multislot user is allocated the channel that has maximum number of free slots. The advantage of this scheme is that it

Arrival rate of Multislot users	0.2
Service rate for Multslot users	150/9.6
Arrival rate Single slot	2.5
Service rate for single slot users	2
Number of channels	8

Table 4.1: Arrival and Service Rates

concentrates the single slot users to a minimal set of frequency channels leaving other channels relatively free for the multislot user.

This can be implemented by broadcasting the number of slots busy in each channel on the logical broadcast channels, and allowing the single slot user to choose the channel which is the busiest and contend for it. The same broadcast message can be used by the multislot user to contend for the most lightly loaded channel. In another approach, the mobile senses all the channels and chooses the least/maximum loaded one without waiting for the base station to provide this information. The callers are dropped when all the slots in all the channels are busy.

The simulations results for this approach are shown in Figure 4.3. The first plot indicates the average number of slots allocated to the MS over time. The X-axis indicates the simulation time. The second plot shows the blocking probability of a Single Slot (SS) user. And, the third plot shows the blocking probability of a Multi Slot (MS) user. We observe that and MS receives 6.78 slots on a average for the arrival rates chosen. The SS users are blocked 2.4% of the times and the MS users are blocked 2.3% of the times. The blocking probabilities obtained above are close to the design objectives of the PACS radio system

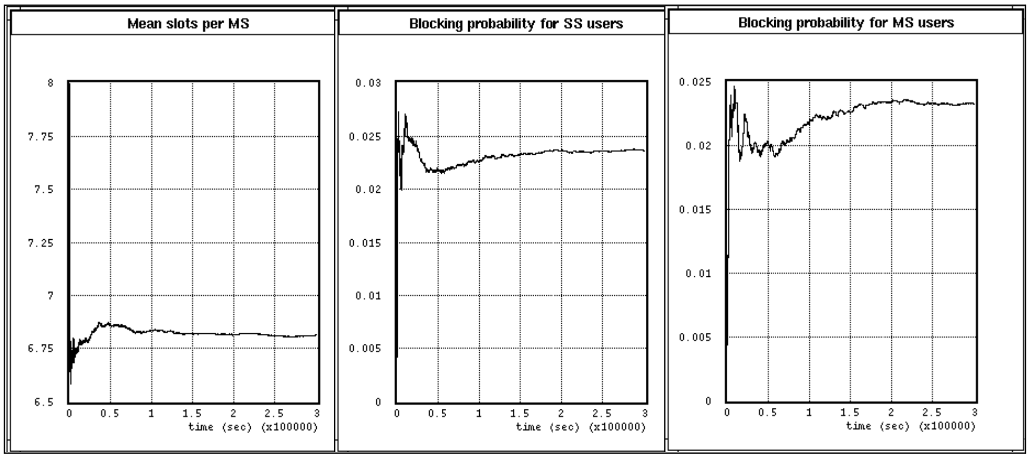

Figure 4.3: Basic FIFO Algorithm

layout. Later, we vary the arrival rates and measure the same statistics to show the effect of the arrival rates. This scheme converges in a short period of time compared to the other schemes.

4.1.2 Reallocation of Single slots users

Another allocation algorithm is based on an optimal reallocation policy for single slot users. Reallocation means moving a user from one frequency channel to another. One such policy is to move a single slot user from the most lightly loaded channel to a heavily loaded channel when a multislot caller arrives. This way a multislot user obtains more slots on the lightly loaded frequency channel.

The SS is not moved if no other channel has free slots.

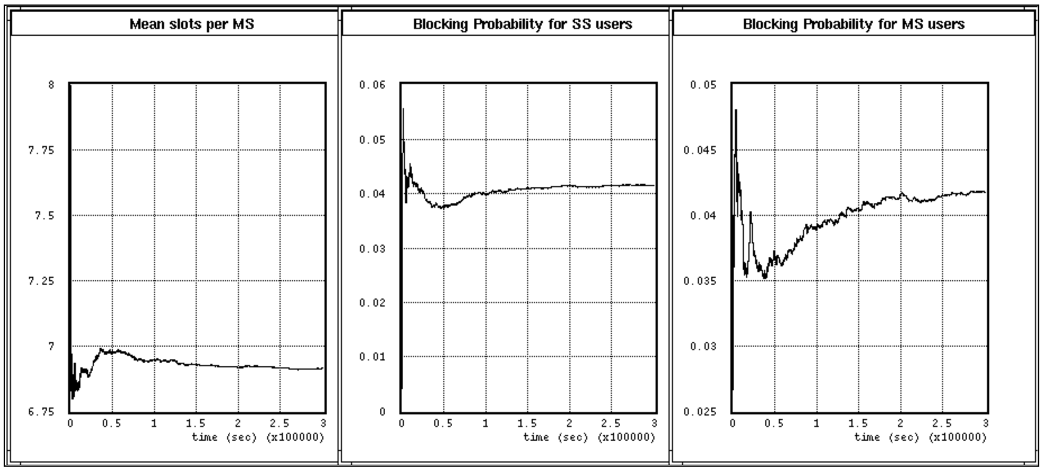

Figure 4.4: Reallocation Algorithms

Reallocation is performed only if the number of busy slots in a lightly loaded channel are below some threshold number. This threshold must be carefully selected and some rule-of-thumb numbers will be deduced from simulations. We chose a value of 6 for the threshold in our simulations.

The reallocation scheme involves using a unicast message from the Base Station to the mobile user, instructing it to change its frequency channel and time slot. The mobiles need the capability built into them to change their frequency channels and time slot with minimum disturbance to the user.

The results for this approach are shown in Figure 4.4. The MS receives 6.85

slots on average. The SS users are blocked 4.0% of the time and the MS users are blocked 4.1% of the time. The percentage increases because more slots are now occupied by the MS users. The increase in the slots per MS over the previous scheme is only 0.07. This makes the policy less appealing.

Reallocation is possible in the present PACS standard. RPCU can force a user (SS or MS) to change its RP to some other RP using the PERFORM_ALT message. ALT is Automatic Link Transfer, which is equivalent to a handoff from one RP to another.

4.1.3 Multi-channel allotment

Instead of moving a single slot user to another channel, the multislot user can send data on different slots on different frequency channels. That is, the multislot user is allocated non-overlapping sets of slots on different frequency channels. The mobile switches between these frequencies within each 8-slot frame.

This system may be limited by the latency in switching between frequencies and the mobile might need to skip a slot when it is switching between the frequencies. This allocation scheme increases the complexity for call setup between the base station and the mobile. The mobile needs the capability to switch between frequencies.

Identifying non-overlapping sets of slots on different frequency channels was acheived by using an $\mathbf{n} \times \mathbf{c}$ matrix. The rows indicate the channels and the columns indicate the slots. The algorithm used was the folowing:

Step 1 The elements in the matrix were marked '0' for available slots, and '-1' for busy slots.

Step 2 Start from the most idle row (max. no. of '0's), all the free columns

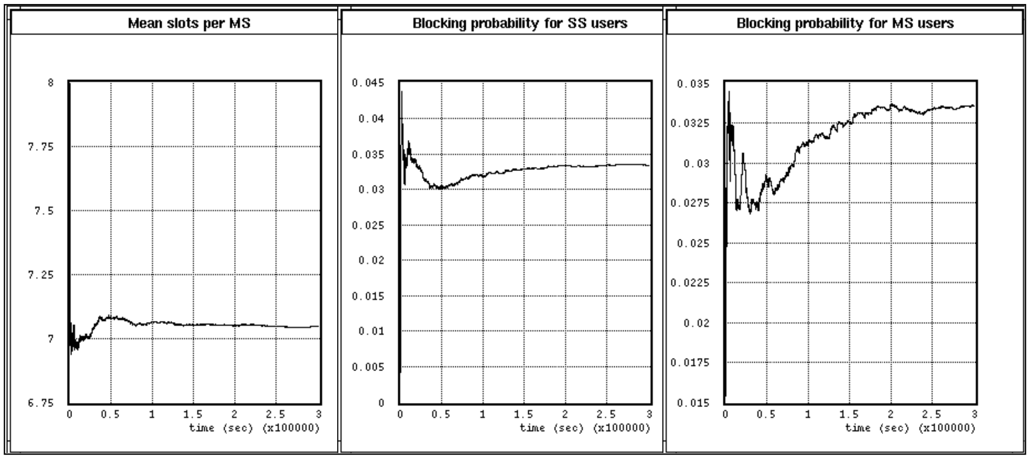

Figure 4.5: Multi-channel Algorithm

in that row were marked '2', indicating that these slots will be allocated to the MS.

Step 3 For each element marked '2' in Step 2, all the elements in the same column of the other rows were marked '-1'.

Step 4 Steps 2 and 3 were repeated till there is no '0' left in the matrix.

Step 5 All elements containing '2' are the chosen slots for the MS.

The results for this approach are shown in Figure 4.5. The MS receives 7.08 slots on average. The SS users are blocked 3.4% of the time and the MS users are blocked 3.35% of the time. This scheme performed better than the previous schemes because it made the best use of available time slots.

4.1.4 Dynamic Addition of slots

An addition to the previous schemes is a dynamic addition of slots to an existing allocation. That is, during a multislot user's call, if a single slot user drops or ends a call, that free slot can be additionally allocated to the multislot user. The dropping SS must be on the channel currently used by the MS.

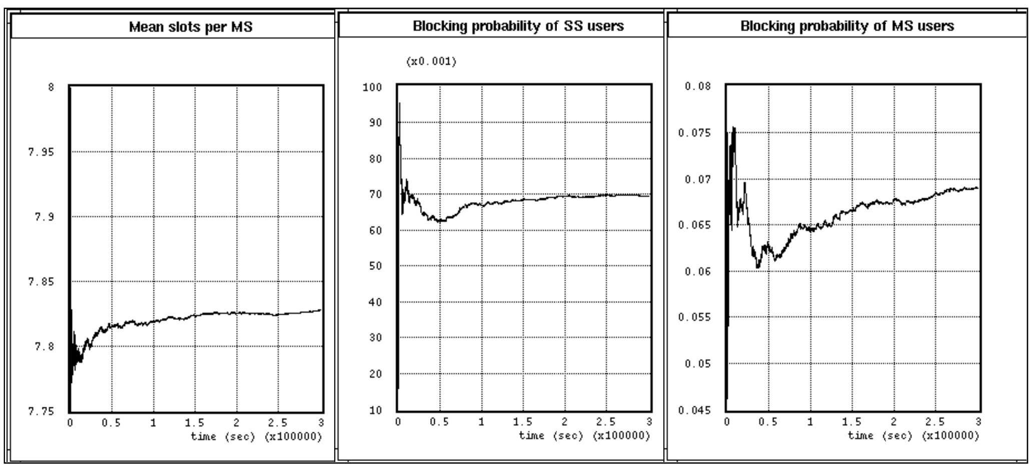

Figure 4.6: Dynamic slot addition algorithms

This scheme adds complexity to both, the base station and the mobile. The multislot user must have an adaptive software to use the additional slot, and generate more data. The base station must be able to add more B channels to the hospital if required. The effectiveness of a new slot depends on how long the ambulance is going to stay in that cell, and how much data is the mobile generating at that particular moment.

The results for this approach are shown in Figure 4.6. The MS receives 7.83 slots on average. The SS users are blocked 7% of the time and the MS users are blocked 6.8% of the time. Again, the increase in the blocking probability is because the MS users are getting more slots on average leaving fewer free slots for arriving MS and SS users. The tremendous increase in the slots per MS is very promising. But, if we take into account the fact that the additional slots are useful to the MS only if it receives them for a long enough period. If the slots get freed as the MS is leaving the cell, the additional slot will not be very useful.

The results shown in Figure 4.6 are for dynamic addition on the reallocation policy. Dynamic addition on Multi-channel allocation is complex because the RPCU must follow all the five steps mentioned in Section 4.1.3 every time an SS leaves.

4.1.5 Reservation

A brute force method is to always reserve one frequency channel for multislot users. This might lead to wastage of the precious resource (bandwidth) if the multislot calls are not frequent enough. So this decision depends on the frequency, or arrival rates of such calls at a base station. If such a reservation is used, the Intelligent Networks signaling will be required because, there will be multiple multislot users in a city and the ambulance will need to know which cells have the channels free and choose their paths accordingly.

In another approach, the base stations could request single slots users to volunteer and release their slots. Some sort of incentive could be provided to them. This is completely a business/political/philosophical decision.

Reservation can be acheived in the present PACS standard by commanding the RPCU to mark a channel solely for priority or emergency users.

4.1.6 Conclusions

We observe from the graphs that the dynamic slot addition when combined with the single slot user reallocation policy, gives the maximum number of channels on average to the ambulance (MS). But the complexity and hence time required to allocate slots is higher than the simpler schemes.

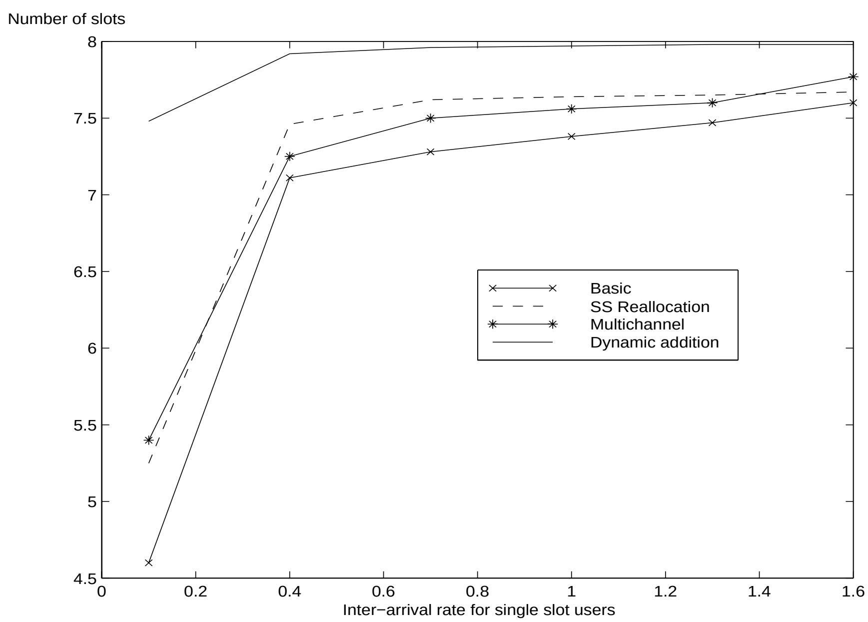

Figure 4.7: Comparison of schemes by varying the SS inter-arrival rates

When the simulations were run on a Sun Ultra 1, the excess time taken to complete a simulation (due to the dynamic slot addition algorithm) was 30 mins for 50,000 arrivals. So, the difference in time for slot allotment per caller is 0.036

sec (= 30*60/50,000). Since this difference in time is not significant compared to the increase in average slots per MS, we believe that the RPCUs must implement the dynamic slot addition scheme in conjunction with reallocation of single slot user. The problem of latency in switching between frequencies makes the Multi-channel allocation an undesired option; otherwise, it performs better than the reallocation scheme.

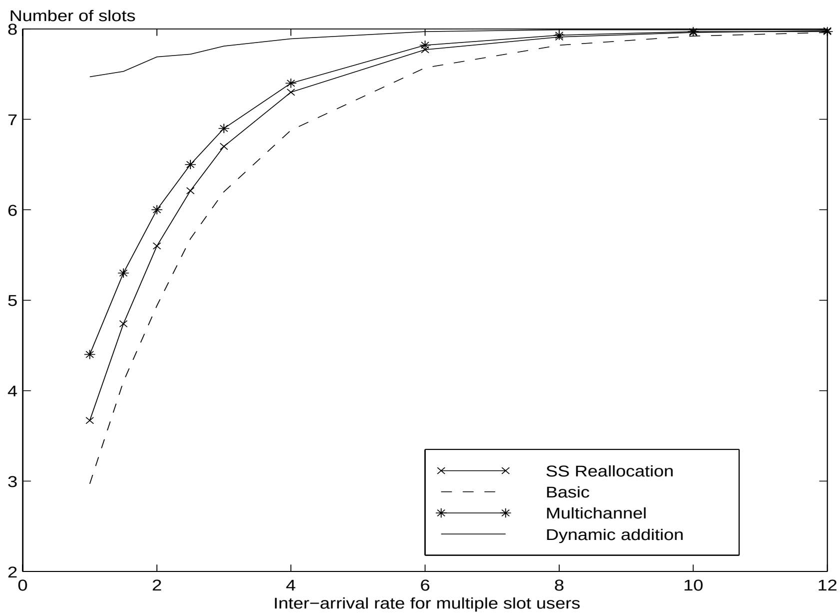

Figure 4.8: Comparison of schemes by varying the MS inter-arrival rates

Figure 4.7 compares all these schemes (in terms of average slots allocated per MS user) when the inter-arrival time of the single slot user is changed. As expected, the slots per MS user increases with decrease in the inter-arrival time. The difference between the schemes becomes less noticeable at low arrival rates. The dynamic slot addition scheme, however, remains at least 0.2 slots above the rest at even reletively high inter-arrival times. Figure 4.8 compares these schemes when the inter-arrival time of the multiple slot user is changed. The results indicate similar trends as in Figure 4.7. All the schemes merge at low arrival rates and become as good as the dynamic slot addition scheme.

We conclude that under low traffic conditions, the simpler techniques perform as good as the more complex ones. So, the implementation should really depend on the traffic expected at system deployment time. The RPCUs can switch between the algorithms when the traffic changes.

4.2 Ambulance Queue Management

As discussed in Chapter 1, the ambulance has various types of sources, generating data with different arrival data rates. The data generated by these sources must be transmitted on the slots that the mobile gets (Figure 4.9). There are several ways in which this can be achieved and each has its own advantages and disadvantages. Since the mobile typically has a very little amount of memory, it must manage the queuing of its packets efficiently. The schemes must also be fair to low rate data, like heart beats and ECG. Some types of data sources are less tolerant to losses than are others. For instance, heart rate must reach correctly and no loss is tolerable. While an occasional loss of a packet may not significantly affect the video and voice quality.

The mobile must have some mechanism to divide the data it is generating into different groups, each to be transmitted on one time slot. There are several ways in which this can be done. Some of these algorithms are simulated in Opnet and two statistics were recorded and compared, namely, the average delay per packet and the packet drops per stream. The latter is done to estimate the fairness of the algorithm with respect to the different data streams.

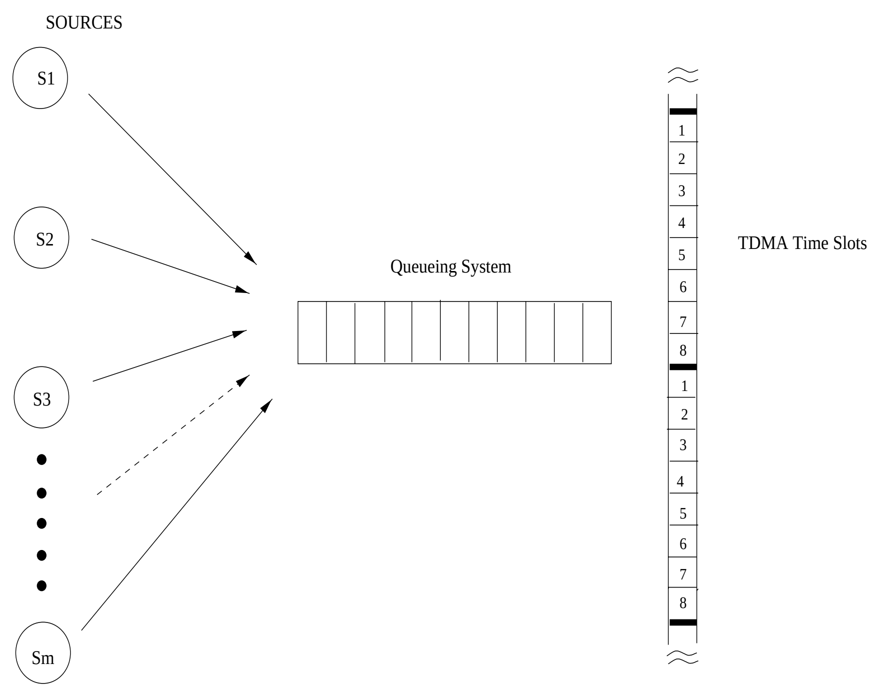

Figure 4.9: Mobile's Queuing System

There were assumed to be five different types of sources, namely, Video, Voice, ECG, Ultrasound and Data. The arrivals (Table 4.2) are assumed to be Poisson. The service rate is constant because each packet is assumed to be of same size and is transmitted in one time slot. The number of time slots are changed for every run of the simulation.

The state diagram of our queuing system model is shown in Figure 4.10. It consists of five states: **init**, which initializes the queues and statistic vectors; **arrival**, which enqueues the arriving packets decided by the specific algorithm; **svc_st** which removes a packet from the head of the queue and transmits it; **idle**, where the system waits for an event to occur; and **clock**, which provides a timing mechanism for implementing the TDM system. The arrows in the diagram define the transition relation between the states. The **ARRIVAL** transition indicates the arrival of a packet from one of the sources. The **TIMER_INTRPT** transition indicates that a timeslot has passed. We used one timeslot as a unit of time to simplify the simulation implementation. The **default** transition occurs when no other transition is enabled. The **((!QUEUE_EMPTY)&&((clock%8)** < **no_slots))** transition indicates that the queue is non-empty and the present clock corresponds to a time slot that has been allocated to the MS. **no_slots** is the number of slots allocated to the MS and **clock%8** is the slot number at any given time instant. Though in reality, the allocated slots need not be consecutive, we assume so in the simulation for simplification.

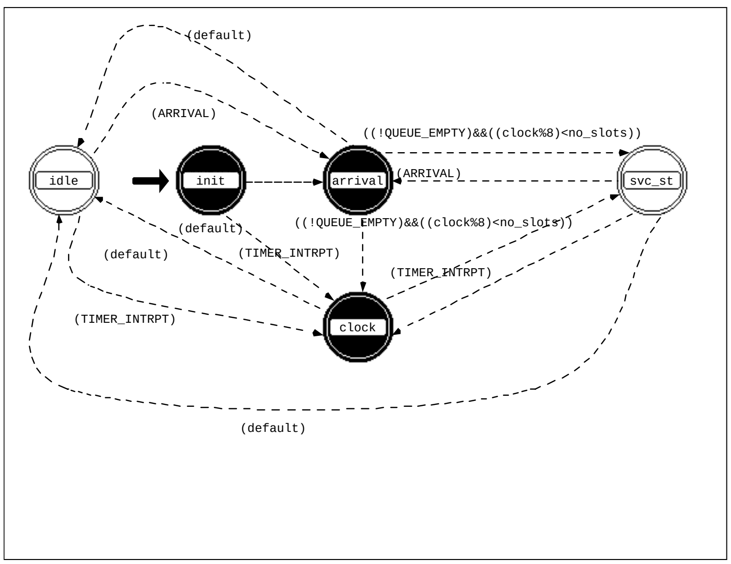

Figure 4.10: State diagram for the mobile queuing system

The algorithms studied are described in the following sections.

4.2.1 Single Queue without Priority

The mobile contains one queue and the packets generated from each source are inserted into that queue. Whenever a time slot that has been allocated to the

Video	1/7
Voice	1/9
ECG	1/13
Ultrasound	1/8
Data	1/15

Table 4.2: Arrival Rates

mobile appears, it removes the first packet in the queue and transmits it. This policy is unfair to the low bit rate applications. For instance, the heartbeat packets (low bit rate) may be dropped if the video packets (high bit rate) have clogged up the queue.

The results are discussed in Section 4.2.5.

4.2.2 Priority Queuing

In this system, the high priority data can replace the already existing low priority data in the queue. The priority of a packet is calculated as follows:

The priority is directly proportional to the waiting time of the packet. The dependence is different for different data types. For example, the heartbeat packet's priority increases faster with time than that of the voice packet. In case of a full queue, an arriving high priority packet can replace a lower priority packet at the tail of the queue.

The results for this and the previous schemes are shown in Figures 4.11, 4.12 and 4.13.

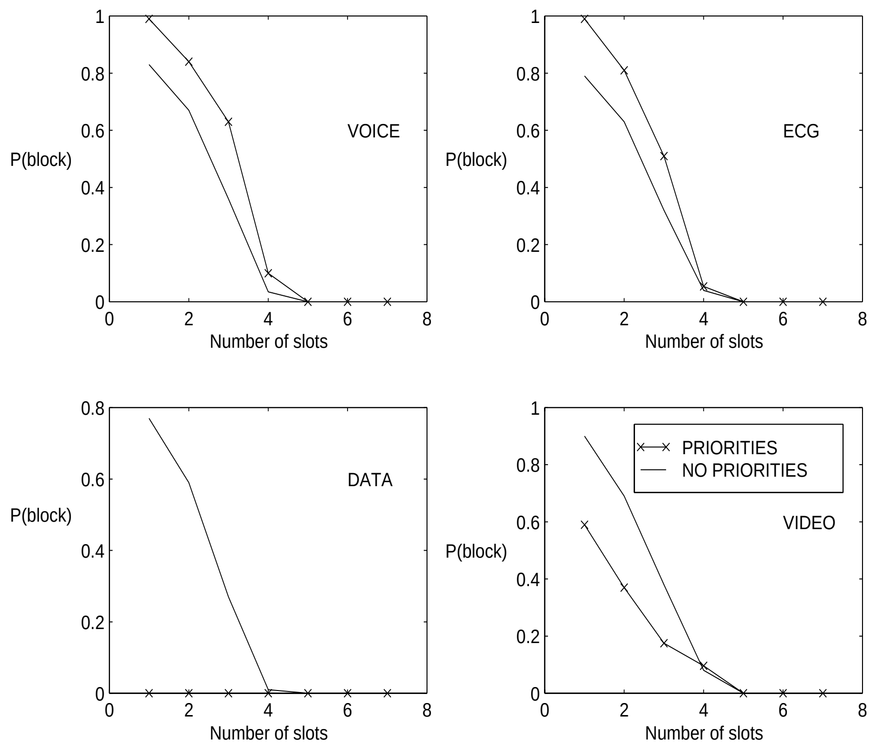

Figure 4.11: Results of Mobile Queuing system

4.2.3 One Queue per Slot

In this scheme, the queue management software maintains one queue per TDMA slot that it receives. Since the amount of memory dedicated for buffering remains the same in all these cases, we need to decide the capacity for each queue. Intuitively, the system will be the fairest when the total arrival rate at any queue is equal to the arrival rates at every other queue (the service rate of all the queues is the same). That is,

if S_i = Set of sources pouring data into the queue of slot i.

$$\sum_{i \in S_1} \lambda_i \cong \sum_{i \in S_2} \lambda_i \cong ... \cong \sum_{i \in S_7} \lambda_i$$

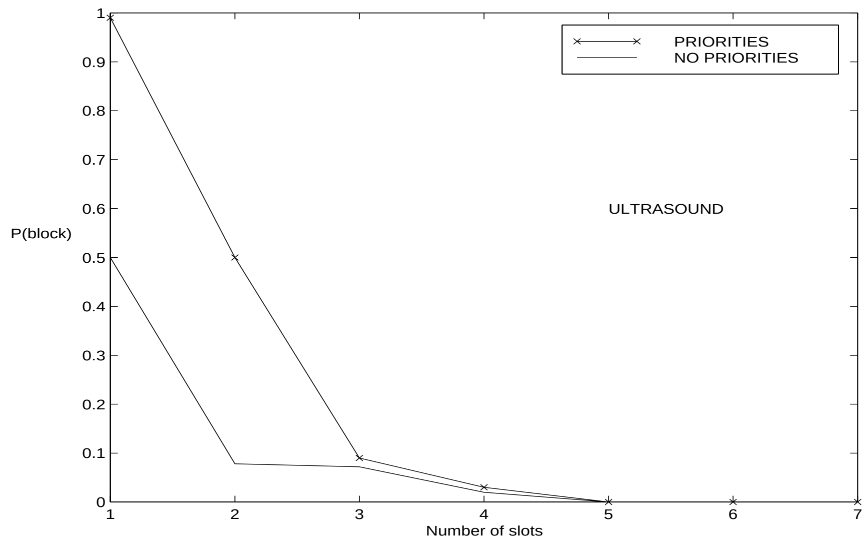

Figure 4.12: Results of Mobile Queuing system

The problem definition is an optimization problem of the following nature:

There are n queues each of length L_i , where i=1,2,...n.

And we need to design configuration of these $L_i's$ that minimizes

$w_1 P(block) + w_2(Avg\_delay)$

where P(block) is the probability of an arriving packet being blocked. This optimization requires use of complex clustering algorithms, to group sources of data.

This scheme become more complex when slots are dynamically added to the ambulance during a call. The clustering algorithm must be run again and the new clusters must be determined.

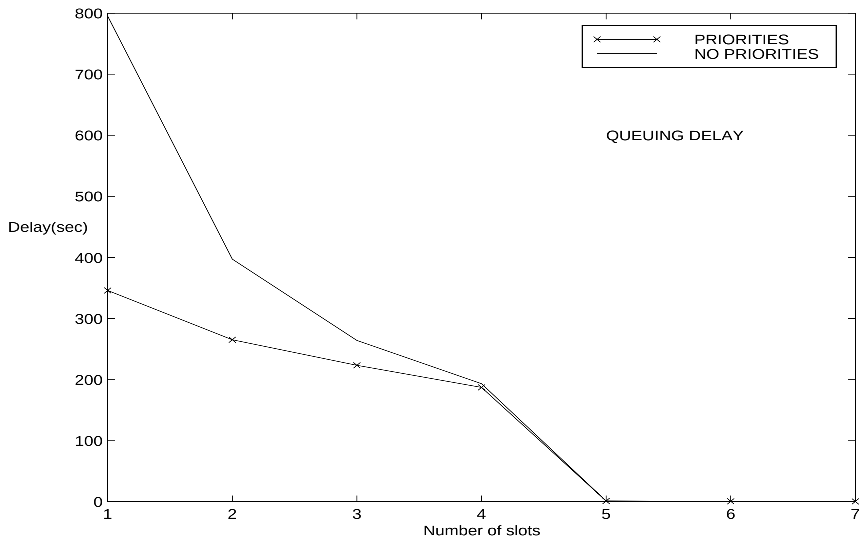

Figure 4.13: Results of Mobile Queuing system

4.2.4 One Queue per data source

This scheme maintains one queue for every data source. In this case the issue is to design the optimal round robin or similar algorithm for serving each queue fairly. This system makes retransmission management more efficient, because each source has its unique characteristics and some of them need retransmissions and some do not. Some can tolerate retransmissions only within a time frame and beyond that time, the packet becomes useless.

4.2.5 Conclusions

Figures 4.11 shows the blocking probabilities for Voice, ECG, Video and Data, Figure 4.12 shows the blocking probabilities for Ultrasound, and Figure 4.13 shows the average delay that each packet suffers.

It is observed from the graphs (Figures 4.11, 4.12, 4.13) that the simple queue has smaller blocking probabilities than the priority queue for Voice, Ultrasound and ECG. For Video and Data, the priority queuing scheme is better. The difference between the performances of the two techniques becomes negligible when more slots are available(more than 5). The average delay per packet is smaller in priority queuing, because a lot of the packets are dropped. Given the importance of the Data source and its delay sensitivity, the priority queue technique is preferred.

Chapter 5

Conclusions and Future Work

In the earlier chapters, we outlined the architecture for development and implementation of our service that provides high bandwidth connection to an ambulance. We discussed several implementation aspects of the new modules designed for this service. In this chapter, we identify the work that can be done in the future to improve the performance of our system.

5.1 Contributions of the Thesis

This thesis proposed a network architecture to provide wireless services in telemedicine. The application is of an ambulance sending medical data (video, ultrasound, ECG, images, X-rays) of a trauma patient over terrestrial wireless channel to the doctors on its way to the hospital. Research has indicated that transmission of critical medical data to the doctors before the patient reaches the hospital, increases the chances of the patients survival by 20%. The ambulance needs a high bandwidth wireless connection to the hospital to be able to transmit the data. In case the bandwidth does not remain constant on its route, it needs to know in advance the approximate bandwidth available to it after every handoff.

This information is used to schedule the transmission of data. For example, the ambulance postpones transmission of video if it knows that it is going to receive higher bandwidth later, or chooses between compression schemes. The ambulance also expects the network to at least reserve the presently free bandwidth for it and avoid new users from using that bandwidth. The ambulance needs to know the best path from the patient's location to the hospital. The best path must provide high bandwidth and take less time.

Presently, ambulances use multiple phones to obtain high bandwidth and have no knowledge of future bandwidth availability. Because each phone makes a separate connection to the hospital, the same data is sent on each line. The ambulances choose the shortest path from the patient's location to the hospital without knowing bandwidth availability on that path. If the call gets dropped, the phone contends for another channel till it receives one.

This thesis attempts to provide a new service to the ambulance with the aid of Intelligent Network (IN) architecture. IN was designed to make fast implementation of new services possible. We design the service for a PACS and IN based network architecture. Instead of designing a distributed algorithm, where the RPCUs would communicate with each other to provide the service, we choose IN, which has a centralized control. This choice avoids making expensive changes in all the RPCUs and uses the simpler IN service deployment platform. Moreover, the centralized control obviates the problem of maintaining relevant databases in each RPCU.

It is not possible to implement this service under the present PACS and IN standards. The changes in these standards were identified and addition of each element was evaluated. We proposed the new Service Independent Building (SIB) blocks required for this service. We proposed the information flows between the IN network elements. We designed two Service Logic Programs (SLP) for executing this service. Our new SLPs included detailed description of the SIBs and the control and data flow sequence. We proposed implementation techniques for each SIB and propose a service independent implementation for them. We proposed the relational database tables required and the queries they must serve. We evaluated our proposed changes in terms of their implementability and complexity. New signaling procedures were suggested which attempt to provide this service with minimum delay and load.

Presence of two kinds of users (single slot and multiple slot) at the RPCU raises issues of designing optimal slot allocation techniques. In the present PACS standard, the slot allocation solely depends on which channel the mobile accesses. In this thesis, we proposed some novel slot allocation algorithms which maximize the bandwidth allocation for telemedicine applications (multiple slot users). We evaluated these schemes through Opnet simulations for various arrival and service rates. We discussed the complexity of each algorithm and suggested implementation techniques for each. We measured the blocking probability for the single slot and multiple slot users and the average slots allocated to a multislot user under each scheme.

The ambulance must multiplex a variety of information sources to be sent to the hospital. Each source has properties that make it unique. The ambulance is limited by the bandwidth it receives from the RPCU and the memory it possesses to queue packets waiting for transmission. We designed and evaluated some queue management schemes at the mobile to optimally manage packets from various data streams. The simulations were performed in Opnet for a variety of data streams with varying arrival rates. We measured packet dropping probabilities, and queuing delays for the packets under each scheme.

5.2 Service Architecture

An optimal protocol needs to be defined between the ambulance and the hospital. Since the data is critical and bandwidth limited; minimum overhead must be used. The packet header should include data type, packet number, checksum. The Mapping SIB must make use of dynamic traffic conditions on the streets before finding the optimal route. These conditions should include construction on the path, busy/idle streets etc. The exact form of database structure to store the SRF information must be studied further to improve query times, for the queries generated by the SCP. For instance, these databases can be implemented in relational, or object oriented or many other forms.

5.3 Slot Allocation Schemes

The schemes evaluated in our work maximize the slots allocated to the ambulance. This may not be the most preferred objective when it comes to optimizing commercial systems where numerous other users are involved and all of them demand higher bandwidth. An all encompassing optimization scheme will include all types of users and their demands. We did not study such a problem in this thesis.

5.4 Mobile Queue Management Schemes

We evaluated two out of the four schemes outlined due to the complex algorithms required in two of the other schemes. The exact priorities for each information source is for further study. The optimal compression algorithms for each kind of information source have not been discussed in our work. It is a topic of extensive research as multimedia wireless services gain market importance.

Bibliography

[1] D. Kim, J. E. C. Jr., and Y. Kim, "Networking Requirements and the Role of Multimedia Systems in Telemedicine," *http://icsl.ee.washington.edu/projects/gsp9/spie1095/*.

[2] Y. Xiao, W. Gaasch, P. Hu, C. Mackenzie, and T. Stair, "Telemedicine and Distributed Decision Making: The Thrill and Issues Beyond," *http://www.hlfrp.ab.umd.edu/telemed.html*.

[3] W. Cobb, "The circuit Switched Solution to Cellular Data," *SPIE Volume 2601/17*, 1995.

[4] Technical Ad Hoc Group 3 of the TI/TIA Joint Technical Committee, *Personal Access Communications System Air Interface Standard : J-STD-014*. Not generally available.

[5] W. Ambrosch, A. Maher, and B. S. (Eds.), *The Intelligent Network: A Join Study by Bell Atlantic, IBM and Seimens.* Springer-Verlag, 1989.

[6] Y.-B. Lin, "PACS Network Signaling Using AIN/ISDN," *IEEE Personal Communications*, June 1997.

[7] J. Norgaard and V. B. Iversen, *Intelligent Networks and New Technologies.* Chapman and Hall, 1996.

[8] T. Hariu, J. Akiba, T. Kishida, and Y. Kondo, "Study on Call Model of Advanced Intelligent Network for Controlling B-ISDN," *IN Workshop* , 1996.

[9] V. Carmagnola, M. Ferretti, and E. Scarrone, "An integrated IN/B-ISDN Reference Architecture for the support of Multimedia services," *IN Workshop* , 1996.

[10] T. Eckardt and T. Magedanz, "Personal Communications Support based on TMN and TINA Concepts," *IN Workshop* , 1996.

[11] K. Takami, H. Sekiguchi, T. Sugiyama, M. Honda, and I. Ushijima, "An Application of Advanced Intelligent Network Technology to Personal Communication Services," *IN Workshop* , 1996.

[12] P. Csurgay, "Service Creation and Deployment in an Intelligent Network Laboratory," Master's thesis, Norwegian Institute of Technology, University of California, Berkeley, CA 94720, May 1987. Memorandum No. UCB/ERL M87/33.

[13] B. Steffen, T. Margaria, A. Claben, and V. Braun, "An Environment for the Creation of Intelligent Network Services," *IEC: IN/AIN Technologies, Operations, Services and Applications - A Comprehensive Report* , 1996.

[14] T. H. Cormen, C. E. Leiserson, and R. H. Rivest, *Introduction to Algorithms.* MIT Press, 1989.

[15] L. Kleinrock, *Queuing Systems. Volumes I and II.* John Wiley and Sons, 1976.