Intelligent Agents in Telecommunication Networks

Costas Tsatsoulis and Leen-Kiat Soh

Information and Telecommunication Technology Center (ITTC)

Department of Electrical Engineering and Computer Science

University of Kansas, Lawrence, KS 66046

Tel: (785)864-7749 Fax: (785)864-0387

E-mail: {tsatsoul, lksoh}@ittc.ukans.edu

Keywords:

Telecommunication networks, intelligent agents, multiagent systems, network management, mobile agents, intelligent interface agents, swarm intelligence, ants, network management architecture, network diagnosis, network traffic control and routing, network mobility platform, network configuration, network monitoring and accounting, service management and provisioning, negotiating agents.

1. Introduction

Telecommunication networks today usually exist in a large, heterogeneous environment. The network components feature different operating systems, platforms, communication languages, and vendors. These components may be incompatible and require a channel or link to facilitate cooperation and coordination within the networks. In addition to such multiplicity in telecommunication networks, we are witnessing a growing enterprise of data, in terms of both demand and supply. The importance and the need for data collection and the subsequent data distribution are becoming, more than ever, paramount. Interactions among networks or network components are inevitable due to the proliferation of data. This phenomenon has brought on a flood of various on-line activities such as advertisement, sale, research, information gathering, information passing, etc., which results in network traffic congestions and strains network management. Subsequently, this has demanded a better handling of the collection, processing, distribution, and understanding of data within telecommunication networks.

When the networks were more constrained and localized, a centralized management approach was adequate to handle various system administration and traffic control tasks.

However, today's telecommunication networks are constantly expanding with many distributed activity centers. This development naturally points to a distributed approach to address issues such as routing, switching, configuration, accounting or monitoring, performance, security, and reliability in telecommunication networks. Instead of one centralized and usually very large system that assumes the complete control and intelligence of the network, a number of smaller systems, or agents, can be used to help manage the network in a cooperative manner. This has motivated the multiagent systems (MAS) in telecommunication networks.

In this chapter, we will present an overview of the application of intelligent agents to telecommunication networks. In Section 2 we discuss what constitutes an agent and its intelligence. In Section 3 we talk about how intelligent agents can help managing telecommunication networks, ranging from increasing the reliability and security of the networks to providing better human-computer interface tools and fault diagnosis. In Section 4 we mention several research projects and approaches that are investigating and exploring new technologies in intelligent agents in telecommunication networks. In Section 5 we discuss negotiation-based communications among agents and how it applies to agents in telecommunication networks. In Section 6 we present a design example of an intelligent agent in telecommunication networks. Finally, we conclude the discussion in Section 7.

2. Agents

In this section, we report on some contemporary taxonomies of agents and some criteria required for agenthood.

Nwana (1996) provided a comprehensive discussion of agent characteristics and typology. The author identified a minimal list of three primary attributes: autonomy, learning, and cooperation. Autonomy refers to the principle that agents can operate on their own without the need for human guidance. For agent systems to be truly smart, agents would have to learn as they react or interact with their external environment. In order to cooperate, agents need to possess a social ability, i.e., the ability to interact with other agents and possibly humans via some communication language. From agents that possess these attributes, the author further distinguished collaborative agents, interface agents, collaborative learning agents, and smart agents. In addition, agents can sometimes be labeled by their applications such as information agents, reactive agents, mobile agents, and hybrid agents.

Wooldridge and Jennings (1995) provided two notions of agenthood. A weak agent enjoys the following properties: autonomy, social ability, reactivity, and pro-activeness. To be reactive or responsive, agents must perceive their environment and respond in a timely fashion to changes that occur in it. To be pro-active, agents do not simply act in response to their environment, but they can also exhibit opportunistic, goal-directed behavior by taking the initiative. A reactive (or responsive), pro-active and social agent is also a flexible agent. A stronger notion of agenthood involves ascribing to agents humanistic concepts such as the belief-desire-intention (BDI) model (Shoham 1993, Rao and Georgeff 1995) and human characteristics such as trust and competence (Maes 1994) or emotions (Bates 1994).

In (Jennings *et al.* 1998), the authors defined an agent as a computer system, situated in some environment, that is capable of flexible autonomous action in order to meet its design objectives. The three key defining concepts are situatedness, autonomy, and flexibility. According to the authors, situatedness means that the agent receives sensory input from its environment and its actions affect the environment. By flexible, the authors indicated that an agent is responsive, pro-active, and social.

In addition, others have identified collaborative interface agents (Lashkari *et al.* 1994), infrastructure agents (Eaton *et al.* 1998), task-specific and performative agents (King 1995) such as search agents, navigation agents, information agents (Papazoglou *et al.* 1992), softbots (Etzioni and Weld 1994, Etzioni *et al.*1995), knowbots (Hylton *et al.* 1996, Hylton and van Rossum 1997), infobots, autobots, webbots, and others. Magedanz *et al.* (1996) taxonomically divided agents into local agents and networked agents for single-agent systems, DAI-based agents and mobile agents for multiagent systems. There are also advisory agents that work as personal assistants to users in email management, news reading, Web browsing, and searching (Lieberman 1995, Maes 1994).

Many attributes sufficient or necessary for defining an agent have been proposed and identified. Here we provide a list of these attributes (which is by no means complete): (1) autonomy–an agent can operate without direct intervention of human users or without having to depend on human users after creation and deployment, (2) communication ability–an agent can interact with other agents or human users to collaborate, negotiate, or coordinate in order to define and perform its tasks, (3) reactivity–an agent can perceive its environment and responds to cues automatically without delay and having to consult its human user, (4) mobility–an agent can

move across operating environments and perform tasks remotely, (5) adaptivity–an agent can learn from its experience in dealing with its environments, including modeling human users, and (6) pro-activity–an agent can reason, plan, and execute tasks by taking the initiative without prompting. Other attributes include (1) veracity—an agent will not knowingly lie (Galliers 1988), (2) benevolence—an agent will carry out its tasks faithfully (Rosenschein and Genesereth 1985), and (3) rationality—an agent will act in order to achieve its goals (Galliers 1988).

Intelligence can be defined as the ability to learn, the ability to adapt, the ability to improve one's performance over time, the ability to reason and make decisions, the ability to act without having to be instructed, the ability to plan and execute a complicated task that involves collaborations from other agents or human users, the ability to traverse a network and perform assigned tasks at each planned stop, and so on. We see an intelligent agent as a piece of program that is mobile, autonomous, reactive, and communicative. Of course, other attributes such as those mentioned above can be structured into the behavior of the program code to increase its intelligence. In our opinion, intelligent agents come in a spectrum of various degrees of intelligence, just like actual life forms in our world that range from single-purpose, single-cell life forms to extremely complex life forms like humans.

Moreover, in our discussion, we do not require agents to be intelligent to exhibit intelligent behavior collectively; that is, single-purpose agents performing a collectively intelligent task are considered, as a whole, an intelligent multiagent system. This is known as swarm intelligence. We discuss this field of intelligent agents in greater detail later in Section 4.1.C of this chapter.

Traditionally, Distributed Artificial Intelligence (DAI) is defined as the research into systems composed of multiple agents, and this field has been divided into Distributed Problem Solving (DPS) and Multiagent Systems (MAS). DPS considers how a particular problem can be solved by a number of modules (nodes), which cooperate in dividing and sharing knowledge about the problem and its evolving solutions. A system of agents is a multiagent system (Lesser 1995) or an agency (Agre and Rosenschein 1996). In contrast to DPS, research in MAS is concerned with the behavior of a collection of possibly pre-existing autonomous agents aiming at solving a given problem. A multiagent system can be defined as a loosely coupled network of problem solvers that work together to solve problems that are beyond the individual capabilities of knowledge of each problem solver. The characteristics of MAS are: (1) incomplete viewpoint of the world for each agent, (2) lack of global system control, (3) decentralization of data, and (4) asynchronous

computation (Jennings *et al.* 1998). Thus, we see that MAS is useful to telecommunication networks due to the implied efficiency, robustness, ability for integration of various systems, and flexibility.

A multiagent system inherits most of the advantages of distributing intelligence over centralized, sequential processing, since it is: (1) reliable – a multiagent system is more fault-tolerant and robust, (2) modular – agents can be added and deleted without greatly disrupting the system; a multiagent system has better scalability, (3) adaptive – agents can re-configure themselves to suit system changes such as noise, resource reallocation, and faults, (4) concurrent – agents can reason and perform tasks in parallel and asynchronously, resulting in faster and flexible execution of tasks, and (5) dynamic – agents can collaborate to form dynamic groups to solve specific problems, pooling together resources, and disband after the problems are solved, releasing resources to local usage.

One of the most important infrastructure for a multiagent system has to do with the communications: protocols and messages, communication channels, communication handling, and mobility of agents. In all implementations of multiagent systems in which agents communicate and travel, these issues must be resolved first before the development of the individual agents. The telecommunication networks currently in place are natural environments for intelligent agents to populate—agents can communicate, be mobile, and perceive detectable environmental cues to react to them. Some of these networks have already in place the fundamentals of the gateways and highways required for intelligent agents to grow and thrive.

3. Intelligent Agents and Telecommunication Networks

One of the most important research areas in telecommunication networks is network management. Network management determines the allocation of network resources (such as data storage, processing power, and memory), fault diagnosis and repair, system administration, routing (switching, bridging, etc.), communications among network components, etc. One of the goals is to have balanced loading and reliable loading on the network such that connections in the network can be established quickly without noise, delay, or numerous trials. In addition, network management also aims at housekeeping the networks so that they work efficiently and effectively, adapt to changes, and respond to problems such as traffic patterns. Thus, in this

section and also in this chapter, we will emphasize network management for telecommunication networks.

Management systems for telecommunication networks usually come in four architectures: (1) centralized network management, (2) hierarchical network management, (3) peer network management, and (4) distributed network management. In a centralized architecture, a single manager handles the housekeeping of the whole network. It checks the network components regularly to ensure the smooth working of the network. It also responds to any warnings and errors issued by the network components. The information and data regarding the network components are stored at a centralized database warehoused by the manager. Thus, the central manager coordinates all network responsibilities from top to bottom. Once a network becomes larger and more complicated, a single centralized management is sometimes not sufficient or efficient. Hence, the hierarchical network management approach is used. In this strategy, managers are arranged in a hierarchy. On top of the hierarchy sits the central manager that administrates a group of assistant managers. Each assistant manager in turn overlooks a group of assistant-assistant managers and so on. In addition, each assistant manager communicates only to its parent manager, with no or minimal same-level interactions. Each manager maintains a localized database with the higher level managers having access to lower level databases. Note that usually these assistant managers are not application- or task-specific; instead, they manage a region of the whole network. Thus, in this architecture, each parent manager coordinates and delegates tasks among its children managers, providing an indirectly cooperative environment among the lower-level managers. In a peer-based network, there exist individual managers that are able to communicate among themselves. Each manager administrates a different domain of the network, interacts with its neighboring managers for information and data, and controls its own database. Hence, there is an increasing sense of cooperation. Finally, in a distributed network, the individual managers are application- or task-specific in terms of their specialties and responsibilities, and each has its own knowledge and databases. Therefore, this architecture requires the highest amount of cooperation from its managers. An extensive survey of network management approaches can be found in (Martin-Flatin and Znaty 1997).

Since intelligent agents come in handy whenever there is a system of coordination and cooperation, the intelligent agent technologies lend readily to the field of telecommunication networks. Intelligent agents in telecommunication networks are becoming more necessary and

feasible because of the current research and development trends are focused towards decentralization and cooperation. This is because today's networks are no longer manageable using just centralized or highly coordinated management strategies because of their size and heterogeneity.

From another viewpoint, the goal of network management technologies is to reduce the risks and cost exposure associated with operations of enterprise systems (Yemini 1993). Administrators or vendors equip their network components with agent software to monitor and collect operational data, such as error statistics, into databases, or to detect unexpected events such as connections overflow or overcrowded traffic. In (Yemini 1993), management platform workstations poll device data, or respond to event alerts sent by the network components. This management paradigm, after the Open Systems Interconnection (OSI)[1] management model, is platform-centered where management applications are centralized at each platform, separated from the database and network components. Each managing platform is using the common management protocol to access managed information provided by an agent residing at a network node. The agent maintains a management information tree database that models a hub using managed objects to represent LANs, interfaces, and ports. A platform can use the protocol to create, delete, retrieve, or change managed objects in the tree model, invoke actions, or receive event notifications. Therefore, a network management system should be able to reconfigure the managed network and respond timely and effectively to monitored events.

According to Sahai *et al*. (1997) a network management system contains four types of components: (1) network management systems, (2) network management programs running on managed nodes, (3) management protocols, and (4) management information. A network management system uses the management protocol to communicate with agents running on the managed nodes. The information passed between the system and the agents is defined by a Management Information Base (MIB). The management standards that have emerged are the Simple Network Management Protocol (SNMP) and the OSI management system that utilizes the Common Management Information Protocol (CMIP). Thus, a network management has the following additional functions: behavior monitoring, installation of components, resource auditing (accounting), information retrieval, and health monitoring.

---

[1] OSI was created by the International Organization for Standardization (ISO) to develop standards for data networking to facilitate multivendor equipment interoperability.

In (Meyer *et al.* 1995), network management is seen as capable of scripting and delegating agents to remote sites where they are incorporated into the local network management program and used for intelligent tasks such as management information base filtering. This application brings mobile agents into network monitoring and network control. Indeed, facilitating the migration or traversal of mobile agents in a telecommunication network allows asynchronous and cooperative processing of tasks, specialization of services, network configuration, decentralization of management, active service usage, intelligent communications such as negotiations among agents, and dynamic information flow.

With a centralized management system, telecommunication networks do not scale gracefully and lack flexibility. This is because of the difficulty in acquiring and maintaining an efficient control and computation with the centralized management system, the requirement that the system communicating to every component of the network, and the distributed and heterogeneous nature of today's network. So, from the standpoint of designing, building, and maintaining a telecommunication network, multiagent approaches are certainly more feasible.

Some standards regarding intelligent telecommunication networks have been proposed: the mobile agent framework of the Open Management Group (OMG) (Cheng and Covaci 1997), the Open System Interconnection (OSI) protocols defined in the 10040 Systems Management Overview by the International Organization for Standardization for data networking, the Telecommunications Management Network defined in M.3010 and the Intelligent Networks defined in Q.1200 by the International Telecommunications Union. These standards have called for more decentralized, distributed network management approaches, with specialized modules such as agents and treating network components as individual entities.

In conclusion, incorporating agent technologies and intelligence into telecommunication networks encourages extensive use of information processing modules, better management and usage of network components and resources, modularization and accounting of network components, flexible configuration of network components (such as self-adapting and self-configuring architecture with plug-and-play capability), and customization and specialization of services.

Some concerns have been raised for using mobile agents in telecommunication networks since these agents influence the performance of the networks. Baldi and Picco (1998) have performed tradeoff experiments regarding mobile agents in telecommunication networks, and the

authors discovered that with the particular goal of optimizing network traffic, the tradeoffs depended on the characteristics of the network being managed (costs, number of nodes, protocols) and of the management task (possibility of local/global semantic compression, expected frequency of invocation, complexity of the task, dimension of replies). In addition, the characteristics of the technology supporting the implementation played a part according to the management protocols (overhead) and the mobile code system used (expressiveness of the language, formats used for transfer, overhead). Hence, according to the authors, whether to use a mobile agent design in place of a traditional client-server architecture requires (1) a model of the management functionality to be implemented, together with the information about the managed network, and (2) a precise quantitative characterization of the management protocols and the mobile code system to be used for the implementation.

4. Applications

In this section, we present several agent technologies in telecommunication networks and mention several projects and research activities that employ intelligent agents in network management architecture, network diagnosis, traffic control and routing, network mobility platform, network configuration, and network monitoring and accounting. These technologies include mobile agents or mobile computing, intelligent interface between agent and human users, swarm intelligence, and economic modeling. Mobile agents play an important role in *spreading* intelligence across networks when they travel. The mobility allows them to be created, deployed, and terminated without disrupting the network configuration. Interface agents model human managers and learn from them how to manage networks. This area of research has not been applied to telecommunication networks directly, but has the potential to automate or assist in the tasks of system administration and network management. Swarm intelligence stems from the work of artificial life in which unintelligent agents work independently or with relative small amount of collaboration to achieve a greater goal that requires intelligence. Then, we briefly touch on designing network management systems after economic models.

4.1. *Agent Technologies in Telecommunication Networks*

A. Mobile Agents

Mobile agents are sometimes known as mobile codes. On the least intelligent definition basis, mobile agents are programs that can be sent to and executed at a remote site. Some agents are deployed to gather information from a remote site, return with the information, and terminate. Some mobile agents, once launched, are able to plan, reason, and carry out tasks on their own and further decide their course of travel across the networks. Some mobile agents travel around the networks and perform itinerary housekeeping. Some mobile agents are thus also known as itinerant agents (Chess *et al.* 1995). Some agents such as *deglets* (Bieszczad and Pagurek 1998) are created with a sole task of sending the identifier of a visited node to the creator and travel by the means of the implemented migration patterns. These agents have also been identified as discovery agents—for finding and establishing connections.

In telecommunication networks, mobile agents are an exciting and increasingly important field. Joshi and Ramesh (1998) pointed out that the advantages of having a facility of mobile agents include mobile computing, remote search and filtering, and real-time production control. Mobile agents enable disconnected operation for clients where persistence of connection is not required for communication between the remote clients and servers. The client generates an agent for performing a remote operation, establishes a connection session, launches it, and terminates the session. When the client reconnects, the agent returns with the outcome of the operation. This reduces the network traffic and also releases the client from the connection, allowing it to establish connections with other sites. In computer-controlled manufacturing and production systems where machines are network nodes that are to be constantly monitored and reconfigured to perform different manufacturing tasks, run time control of machines require real-time management and communication latency is not acceptable. In such case, agents could be created with the control itinerary and dispatched across the network to remotely control the machines in real-time.

Baldi *et al.* (1997) further pointed out that the approaches proposed in the Internet Engineering Task Force and the International Organization for Standardization are of low degrees of flexibility and reconfigurability even though these research directions advocated agents for management. This is because the agents are direct extensions of the centralized management and do not have mobility. Baldi *et al.* (1997) showed that agent or code mobility is

essential in enabling a better use of bandwidth resources and a higher degree of flexibility and reconfigurability in telecommunication networks.

In general, in order to manifest agent mobility for network management, a system has to incorporate a mobility framework (Bieszczad *et al.* 1998b). The framework has to support various agent models such as the life-cycle model, computational model, security model, communication model, and navigation model. For the life-cycle model, services have to be provided to allow the creation, termination, deployment, and suspension of agents. In our opinion, in addition to the above activities, one should consider the reconfiguration of agents as part of the life cycle, enabling agents to learn and adapt as they *live*. This would give the agents more intelligence and encourage knowledge flow, either among agents, or between agents and network managers. The computational model refers to the computational capabilities of an agent, which include data manipulation and thread control primitives. The security model describes the ways in which agents can access network resources, as well as the ways of accessing the internals of the agents from the network. The communication model defines the communication between agents and between an agent and other network components. Finally, the navigation model defines how an agent should travel or reside, or how an agent should be delegated by its launcher.

Other important design issues in telecommunication networks housing mobile agents include transfer protocols, resource allocation, network component access, and user mobility.

Mobile agent technologies are sometimes known as mobile code technologies. An extensive survey and insightful analysis of mobile code technologies such as mobility mechanisms, including mobile code design paradigms and applications, can be found in (Fuggetta *et al.* 1998).

Hierarchical Agents

Appleby and Steward (1994) of British Telecom devised a distributed control mechanism for mobile agents to optimize routing in telecommunication networks according to a least cost criterion. There are two hierarchical levels of agents: load management agents and parent agents. A load management agent is launched on a particular node. The agent visits every node in the network efficiently using an adaptation of Dijkstra's shortest path algorithm, records the current spare capacity, and amends the routing tables. Hence, it collects the necessary information to determine the optimal routes from all other nodes to that particular node on which

the agent has been launched. The parent agents control the next management level. They travel around the network and launch load agents where network management is needed. The decision to launch a load agent is made on the basis of information gathered, and a set of heuristic rules.

B. Intelligent Interface between Agents and Human Users

Intelligent interface agents interface with humans or other agents and learn through solicitation or vigilance (Chin 1991, Dent *et al.* 1992, Maes and Kozierok 1993, Sheth and Maes 1993, Lashkari *et al.* 1994). Intelligent interface agents can learn by (1) monitoring and imitating the human user, (2) receiving direct commands from the user, and (3) receiving feedback from the user. Sometimes, agents can learn from each other through knowledge transfer as well. We think that intelligent interface agent technology is an important aspect of knowledge-transfer between human system administrators or network engineers and software agents. It helps model the critical, diagnostic reasoning capabilities of humans in managing telecommunication networks.

There is little work using interface agents in telecommunications, but we foresee this as a very viable area of future development. Some work on using interface agents for network supervision is described in (Esfandiari *et al.* 1996).

C. Swarm Intelligence

The field of artificial life has given rise to swarm intelligence (Beni and Wang 1989), in which a group of unintelligent agents of limited capabilities exhibiting collectively intelligent behavior (White and Pagurek 1998, Bonabeau *et al.* 1999). Swarm agents are mobile agents, but with much less computational capability and intelligence—they traverse the network carrying out simple-purpose tasks with no explicit knowledge of the ultimate goal or network scenario, and without direct communications or contacts with other agents. However, the collective work of swarm agents can have global and intelligent impact on the whole telecommunication network. Usually these agents are distributed and highly adaptive to changes in the network and traffic patterns and can adapt to the network topology, the call probabilities of the nodes, and temporary situations caused by the randomness of the call patterns.

White *et al.* (1998a) described how multiple interacting swarms of adaptive mobile agents could be used to locate faults in networks. The authors proposed the use of mobile agents for

fault finding in order to address the issues of client/server approaches to network management and control, such as scalability and the difficulties associated with maintaining an accurate view of the network. The authors further defined three principal types of mobile agents: servlets, deglets, and netlets. Servlets are extensions or upgrades to servers that stay resident as integral parts of those servers. Mobile agents constituting servlets are sent from one component to another and are installed as code extensions at the destination component. Deglets are mobile agents that are delegated to perform a specific task and generally migrate within a limited region of the network for a short period of time, for example, to undertake a provisioning activity on a network component. Netlets are mobile agents that provide predefined functionality on a permanent basis and circulate within the network continuously. These agents are small and mobile, and communication is top-down, instead of peer-based.

Ants

The intelligent behavior resulted from indirect communication between agents is called stigmergy (White *et al.* 1998a) and there are two forms of it. Sematectonic stigmergy involves a change in the physical characteristics of the environment. Ant nest building is an example of this form of communication in that an ant observes a structure developing and adds to it. The second form of stigmergy is sign-based. Here, something is deposited in the environment that makes no direct contribution to the task being undertaken but is used to influence subsequent task related behavior such as the foraging behavior of ants. Thus, some swarm systems in telecommunication networks are also known as ant systems.

Schoonderwoerd *et al.* (1996, 1997) designed an ant system for telecommunication networks after the trail laying ability of ants. Ants lay chemicals (pheromones) as they traverse trails. By creating small, mobile agents that are capable of traversing the network themselves and leaving behind simulated pheromones, agents coming after them can gather vital information about the paths and the network as a whole. In this research work, a simulated network models a typical distribution of calls between nodes; nodes carrying an excess of traffic can become congested, causing calls to be lost. The network supports a group of ants that have simple-purposed tasks, with no direct communication capability or explicit knowledge of the global goal. The ants move across the network between randomly chosen pairs of nodes; as they move they deposit simulated pheromones as a function of two variables: (1) the distance from their source node and

(2) the congestion encountered on their journey. They select their path at each intermediate node according to the distribution of simulated pheromones at that node. Calls between nodes are routed as a function of the pheromone distributions at each intermediate node. The ant-based control system was shown to result in fewer call failures than other methods such as fixed shortest-path routing and algorithmic mobile agent (instruction-specific), while exhibiting many attractive features of distributed control.

Other work in ants in telecommunication networks for routing and load balancing has concentrated on improving the path-finding intelligence of ants using dynamic programming (Bonabeau *et al.* 1998), genetic algorithms (White *et al.* 1998c), and Q-learning algorithms (Gambardella and Dorigo 1995).

D. Economic Models

Economic models can serve as the motivation and navigation guidance of intelligent agents traversing a telecommunication networks. The distributed and emergent behaviors subscribed by the models are comparable to those of today's network components. Therefore we foresee many more such models being used in the future when designing multiagent systems in telecommunication networks.

Ferguson *et al.* (1996) used an economic model for intelligent agents in telecommunication networks. According to the authors, computer networks are being used by a growing and increasingly heterogeneous set of network components (computers, channels, and users) that have diverse Quality of Service (QoS) requirements. To support the diversity in large distributed networks, the tasks of efficient service provisioning and optimal resource allocation become very complex but can be accomplished via decentralization. In an economy, decentralization is provided by the fact that economic models consist of agents which selfishly attempt to achieve their goals. There are two types of economic agents, suppliers and consumers. A consumer attempts to optimize its individual performance criteria by obtaining the resources it requires, and is not concerned with system-wide performance. A supplier allocates its individual resources to consumers. A supplier's sole goal is to optimize its individual satisfaction (profit) derived from its choice of resource allocations to consumers. So in a provisioning telecommunication network, the economic model lends insights to the demand and supply

balance, pricing strategies and mechanisms, decision making in choosing suppliers, system economy, and resource allocation.

4.2. Research Areas in Telecommunication Networks for Intelligent Agents

The following discussion is by no means inclusive of all research projects that apply intelligent agents to telecommunication networks. Most of these applications perform more than one single network management task. Therefore, the categorization is not exclusive either. We introduce the categorization to present several important areas in telecommunication networks where intelligent agent technologies are active.

A. Network Management Architecture

In (Sahai *et al.* 1997, Sahai and Morin 1998), a mobile agent environment for network management was described. The MAGENTA (Mobile AGENT for Administration) environment was designed to achieve dynamic and decentralized management of a distributed system comprising of heterogeneous machines running varied operating systems connected by LANs. The environment introduces the idea of a Mobile Network Manager (Sahai *et al.* 1997), equipped with autonomy, reactivity, proactivity, and communication skills. The environment also provides in its architecture *lieus* as places or locations where an agent can originate, reside, execute, and interact with the system as well as other agents. A lieu is a static program that checks the security, authenticates the suitability for execution, allows communications with other agents, provides residence, and keeps track of its agents. Thus, in this management architecture, lieus are static agents and they are capable of spawning and servicing mobile agents that traverse across the network.

Frei and Faltings (1998) proposed an abstraction approach to represent the original network as a hierarchy of simplified graphs. Each node of the graph abstracts a part of the network inside which routing of demands requiring a given amount of bandwidth is possible. The management of this virtual architecture and the routing of demands can then be distributed to intelligent agents.

B. Network Diagnosis

Mobile agents can be delegated to measure utilization and efficiency of nodes throughout the network. If the measurements violate some established network rules or fall below some expected network performance criteria, then the mobile agents have located a fault. Mobile agents can then perform a diagnosis to locate the source of the fault, report the problematic area to human managers, repair the fault itself, or transfer the responsibility to other agents. White *et al.* (1998a) used ants to perform network diagnosis by populating the network with a group of small agents to collect and tally observations which can be checked against diagnostic rules to determine faults.

C. Network Traffic Control and Routing

El-Darieby and Bieszczad (1999) proposed a system of intelligent mobile agents to reduce management traffic on the network (no bandwidth-exhaustive Client/Server message exchanges), enable more robust response to problems (timely, intelligent problem-solving behavior of the agents), reduce administrative overhead and cost due to function delegation (agents perform tasks autonomously), and allow low-level problems to be dealt with locally at the network component (reducing both the processing load on the manager computer and the traffic carrying instructions and data regarding the problems).

Other routing works include White *et al.* (1998b) in connection management and Schoonderwoerd *et al.* (1996, 1997) in network load balancing.

D. Network Mobility Platform (Mobile Code Languages)

Some research has examined guiding and enabling the navigation or migration of mobile agents in telecommunication networks (Acharya *et al.* 1997, Peine and Stolpmann 1997, Peine 1997, Gray 1996, Hylton *et al.* 1996, Straßer *et al.* 1996, White 1996, Lingnau *et al.* 1995). Issues integral to the mobility of agents in heterogeneous networks include code mobility technologies such as portability, execution, security, and resource access.

For example, Ara (Agents for Remote Action) is a platform for mobile agents designed for the portable and secure execution of mobile agents written in various interpreted languages on top of a common run-time core in an attempt to achieve location transparency. It allows agents to migrate at any point in their execution, fully preserving their states, and exchange messages

with other agents.   A network system can contain many virtual places, each establishing a domain of logically related services under a common security policy governing all agents at this place. Agents are equipped with allowances limiting their resource accesses, both globally per agent lifetime and locally per place. The Ara project emphasizes the system support for general mobile agents with minimal features for applications and behaviors of agents (Peine and Stolpmann 1997, Peine 1997).

E. Network Configuration

When network components are added, or suspended (for repair), or deleted from the network, the network reconfigures. Mobile agents can be used to reconfigure the networks and update the network topology. For example, agents can be designed to implement plug-and-play network components (Bieszczad *et al.* 1998a, Yemini 1993) in which components can either announce their existence by self-bootstrapping (sending out agents to inform others) or issue signals to be discovered by network-patrolling agents. New network services can also be provisioned using agents (Csurgay *et al.* 1997, Barr *et al.* 1993).

Pagurek *et al.* (1998) introduced an alternative to existing Asynchronous Transfer Mode (ATM) configuration management solutions. The authors designed and implemented a generic model based on mobile agents, that performs permanent virtual connections (PVCs) as configuration management functions in multi-vendor ATM networks. With the mobile code approach one operator was required to enter the end-to-end PVC configuration requirements (such as port connections for neighboring switches and bandwidth) and the PVC configuration application then sent a PVC agent to the network to conduct the configuration task, with the user requirements carried by the agent. The agent configured the switch at the source, migrated to the carrier's switch and configured it, and finished the task at the destination switch. In contrast to manager-to-manager PVC configuration management and CORBA-based PVC configuration management, the mobile agent approach has several advantages: there is no manager-to-manager software integration; understanding and knowledge about different switches are not required; a PVC agent automates the connectivity procedures and bypasses human decision making; and, PVC agents are task-specific and can be implemented more efficiently.

F. Network Monitoring and Accounting

Network monitoring and accounting are necessary to measure the performance of networks so that networks can be designed and improved over time for performance, efficiency, and cost. However, due to network delays the use of a centralized server makes measurements inaccurate or not timely, leading to network performance measurements are not reliable or difficult to collect.   Instead of remotely polling network components, mobile agents can be dispatched to perform local analyses remotely (Bieszczad and Pagurek 1998). This improves the accuracy of the information, does not consume local resources permanently because of the agents' mobility, and makes the maintenance of the monitoring and accounting system easier since mobile agents are smaller and distributed in design and implementation.

Bieszczad and Pagurek (1998) also pointed out that mobile agents can be used to implement hot-swapping technologies (to keep stationary network monitoring agents up to date) and server migration (information analysis such as service demand, network load, and failure rate). The authors subsequently warned against agent flooding, that is, uncontrolled mobile agents populating the network and taking over a large proportion of its resources, affecting the network performance that is to be measured.

G. Service Management and Provisioning

As pointed out in (Plu 1998), the goal of service provisioning in telecommunications is to allow corporations to provide more and more sophisticated information and electronic commerce services to many customers. Plu (1998) went on to describe several attributes of a service-based agent such as autonomy, trustworthiness, distributed, social ability, persistence and the ability to act asynchronously, data encapsulation, interoperability, and flexibility. In addition, the author prescribed policies to govern performative actions such as obligation, permission, and prohibition.

Other agent-based systems in service management include service maintenance agents and customer agents (Weihmayer and Tan 1992, Busuoic and Griffiths 1993, Weihmayer and Velthuijsen 1998) and operator assistance agents (Garijo and Hoffmann 1992).

5. Negotiating Agents

Cooperative and self-interested agents within a community communicate with each other to achieve a mutually acceptable agreement towards certain goals through negotiations. Negotiation protocols allow network components to conduct intelligent communications to either selfishly increase the performance of each component, or altruistically improve the overall effectiveness and efficiency of the network. The process can be seen as a distributed search through a space of potential agreements (Laasri et al. 1992). Traditionally, negotiating agents have used one of four techniques: game theory (Osborne and Rubinstein 1994, Rosenchein and Genesereth 1985), approaches inspired by operations research (Kraus *et al.* 1995, Sandholm and Lesser 1995), probabilistic/Bayesian techniques (Zeng and Sycara 1996), and heuristic models (Kraus *et al.* 1991).

Game-theoretic approaches usually make assumptions, such as shared prior probability estimates, common knowledge of agents' preferences, and other aspects of the game, and perfect rationality. Operation research approaches usually involve decentralization of computational algorithms. Bayesian approaches update the knowledge and belief that each agent has about the environment and other agents and naturally models the iterative negotiation behavior. Heuristic negotiation techniques such as case-based reasoning in PERSUADER (Sycara 1990a; 1990b) allow the agent to use any set of previously successful algorithms, without having to ascribe to one methodology.

Negotiations are important in multiagent systems in terms of cost reduction, performance optimization, cost budgeting, and autonomy. Negotiating agents have been used to detect and resolve certain kinds of feature interactions that occur in telecommunication systems (Griffeth and Velthuijsen 1993). This research area has the potential to contribute to intelligent agents in telecommunication networks. Readers are pointed to (Rosenschein and Zlotkin 1994) for further discussion on negotiation among agents.

6. A Design Example of An Intelligent Agent Structure

There are numerous designs of agent structure in the literature. This is due to various functions that the agent technologies apply in the area of telecommunication networks, as described above. In this section, we present a design of one such intelligent agent in the context of telecommunication networks. The following discussion is aimed to provide a general idea on

how one such agent structure can be designed and, therefore, some issues related to telecommunication networks are only briefly touched upon.

Figure 1 shows a general design of an intelligent agent. There are six integral modules within the agent: (1) Event Monitor, (2) House Keeping, (3) Adaptive Learning Mechanism, (4) Message Interface, (5) Task Interface, and (6) Manager. The network support for the creation of an agent is provided through the Network Management module.
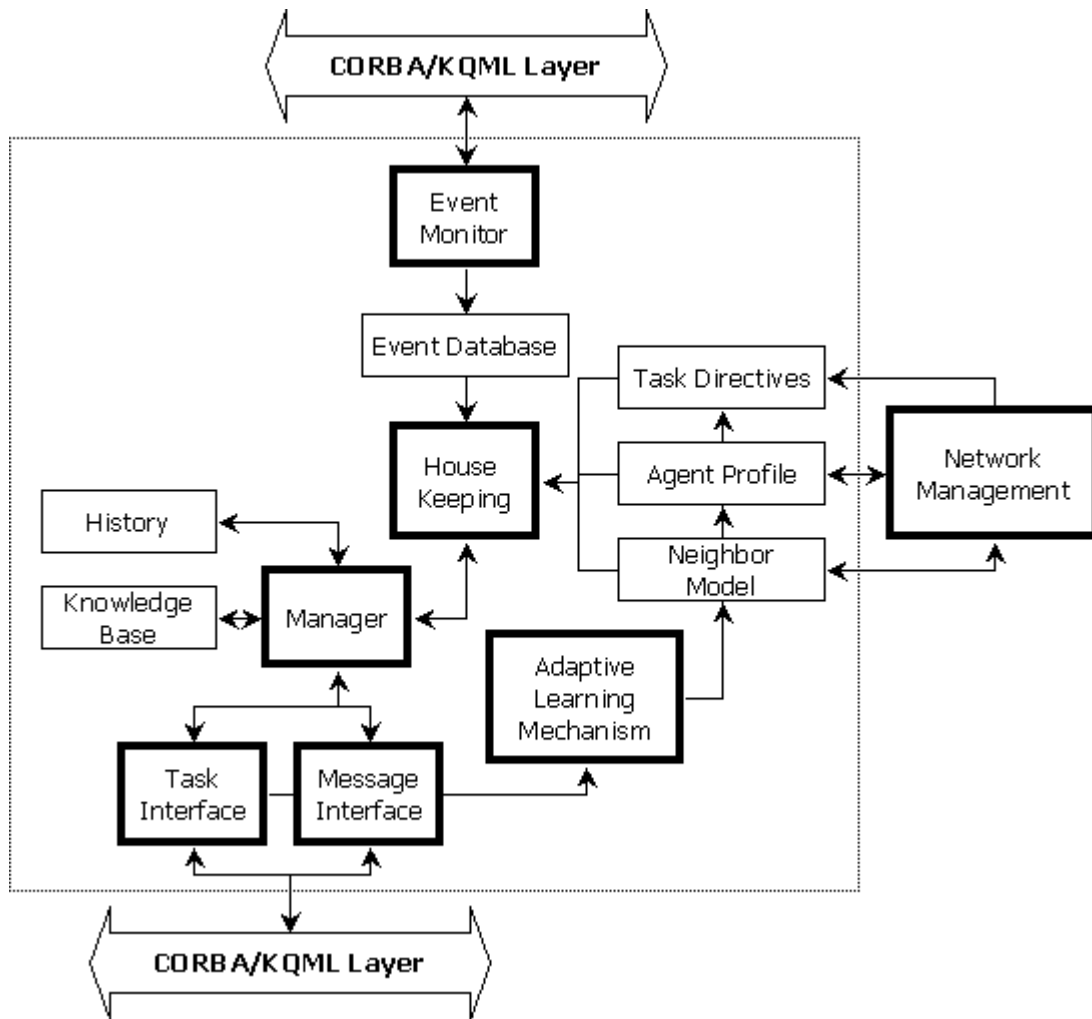

Figure 1 A design example of an intelligent agent structure for telecommunication networks

At the creation of an agent, the Network Management module supplies the agent with its profile, its model of neighboring agents, and a set of task directives. The Agent Profile includes a set of parameterized attributes such as *name*, *origin*, *priority*, *attitude*, *category*, *class*, *role*, etc. These attributes will later be used in guiding the behavior of the agent in task planning, decision making, and interactions with neighboring agents.   The Neighbor Model includes a list of

agents known to the agent. Each neighbor model will include the *name*, *category*, *class*, *role*, etc. Note that we can either provide each agent with a set of complete knowledge of its neighbors, or a partial set. The more complete the knowledge, the more is required of the agent to maintain its knowledge base for decision making. The Task Directives describe the responsibilities of the agent. The *role* of an agent can be a Message Router, a Network Manager, a Resource Allocation Manager (such as computer, data, and information resources), a Navigator, a Network Maintainer, or a Network Housekeeper.

At the creation of an agent, the Network Management module will specify the role of the agent. Along with that specification is a set of task derivatives that outline the set of responsibilities of the agent. The Derivatives will specify the mobility (or residency), navigation plan, deployment plan, and capabilities of the agent, and, most of all, the tasks to be carried out by the agent. For example, a Navigator agent may be created to obtain the fastest route between a network node A in New York and a network node B in Lawrence, Kansas. Thus, this agent is mobile. The agent's deployment plan includes the times that it must be deployed and returned. The capabilities specify what resources the agent has in carrying out its tasks: query about a particular node's security and consistency, measurement or estimation of a node's traffic activity, logging of a node's activity, message sending to communicate with the originator of the agent, etc. The tasks, in this case, will describe several requirements: (1) whether the fastest route is for data flow (and its degree) or computer resource usage (and its degree), (2) the weights on the routes (speed, security, reliability, etc.), (3) whether the selected route is for short or long term usage, (4) a set of steps specifying the tasks. The set of task steps will include what procedures the agent must take when arriving at, say, a node. For example: (a) request for N computational resources, (b) measure node activity, (c) if the node activity is below a certain level, then accept it, (d) select another node down the route (based on the agent's current status, history, and knowledge base), and (e) move to the selected node.

The Event Monitor module interfaces with the Network Environments. This includes alerts or messages sent to the agent by users or system administrators of the networks: for example, a network shutdown of a particular node, network congestion, or even a temporary exclusive usage of a network area. The Event Monitor will log these events in the Event Database, which it will maintain and update for consistency and accuracy.

The House Keeping module fuses the events, tasks derivatives, agent profile, and the neighbor model together to provide the Manager module a modified but consistent set of information, making the Manager temporal and situational aware. The House Keeping module is essentially the information guru of the agent.

The Adaptive Learning Mechanism module allows the agent to learn from its interactions with other network elements. So, as the agent performs tasks during its life cycle, it will be able to learn the behavior of the neighboring agents or network nodes that it encounters. Of course, if the agent is short-lived, then we might want to tune down the learning ability. On the other hand, if the agent is an information-gathering agent, or has a long life cycle (during which it might encounter the same neighbor more than once), then this learning mechanism allows the agent to adapt. The Manager will thus be able to make decisions on more updated and accurate information.

The Message Interface module has two basic responsibilities: (1) send a message to a destination, and (2) receive a message. It will thus know how to compose a message and parse a message. The could be sent across a CORBA layer.

The Task Interface module performs the task. In the Navigator example, this module might have to ask for permission from the local node to perform a task. If the permission is granted, then the module runs its procedures to, say, collect information from the node. In another example, suppose that the agent is deployed and resides at a local node, it may have access to the local computer resources. Thus, in this case, the Task Interface module simply actuates the steps of task in the agent's directives.

The Manager module will be responsible for the decision making and planning of the agent. In order to be intelligent, it has a knowledge base. This knowledge base may consist of rules, features, weights, vectors, etc. and provide the reasoning process to the Manager. The Manager will decide whether, when, and how to carry out a task. It will also update its History base where past behaviors and observations are stored. For example, in the Navigator example, if the Manager realizes that it has found a fastest but also least reliable path, it might have to compute the utility of selecting this path based on its knowledge and information. If there is a rule that says "If the success of the transfer of data if of the HIGHEST priority, then it might be better to choose a more reliable albeit slower path", then the Manager may choose another safer path.

Note that even though we have used an example in which the agent is a Navigator, deployed to map out a fastest path between two nodes, we can extend the above discussion to other intelligent agent roles in a telecommunication networks such as (1) a Message Router – an agent that resides at a single node and distributes and dispatches message efficiently, (2) a Resource Manager – a resident agent that allocates its computer resource or data resource to other network functions or agents, (3) a Network Maintainer – a resident agent that analyzes processes on a node and decides which to terminate (such as hung processes) and which to maintain, and (4) a Network Housekeeper – an agent deployed to collect data on the activity and health of a particular network and terminated once it reports the findings.

7. Conclusions

We have presented an overview of the application of intelligent agents to telecommunication networks. We have discussed the properties of an agent and its intelligence. Again, we have not attempted to define what constitutes an agent and its intelligence. We have talked about how intelligent agents can be of help in managing telecommunication networks in various aspects such as reliability, security, control, routing, and fault diagnosis. In addition, we have discussed agent technologies and research areas in telecommunication networks and cited a number of contemporary research activities around the world. Moreover, we have included agent negotiations as closely related disciplines in intelligent agents to telecommunication networks. We have recommended this branch of technologies be involved in telecommunication networks because we believe that negotiations are key ingredients in designing cooperative and competitive intelligent agents. We have also presented a design example for an intelligent agent in the telecommunication networks context.

To conclude, we see intelligent agents in telecommunication networks as a research area that has great potentials in fusing previous works and experiences of other disciplines such as AI (e.g., DAI and MAS), network engineering, distributed processing, economy, social studies, information technology, data mining, knowledge engineering, system control and management, software engineering, computer languages, operations research, artificial life, psychology, and cognitive modeling. From the AI perspective, we believe that if we inject enough intelligence

Computational Intelligence in Telecommunications Networks, W. Pedrycz and A.V. Vasilakos (Eds.), CRC Press, 2000

into telecommunication networks, then they will become self-sustaining, self-organizing, self-managing intelligent communities in the future.

8. References

Acharya, A., M. Ranganathan, and J. Saltz (1997). Sumatra: A Language for Resource-Aware Mobile Programs, in *Mobile Object Systems: Towards the Programmable Internet*, J. Vitek and C. Tschudin (eds.), Springer-Verlag Lecture Notes in Computer Science, 111-130.

Agre, P. E. and S. J. Rosenschein (eds.) (1996). *Computational Theories of Interaction and Agency*, Boston, Massachusetts: MIT Press.

Appleby, S. and S. Steward (1994). Mobile Software Agents for Control in Telecommunications Networks, *BT Technology Journal*, **12**(2):104-113.

Baldi, M., S. Gai and G. P. Picco (1997). Exploiting Code Mobility in Decentralized and Flexible Network Management, in *Proceedings of First International Workshop on Mobile Agents (MA'97)*, April, Berlin, Germany, 13-26.

Baldi, M. and G. P. Picco (1998). Evaluating the Tradeoffs of Mobile Code Paradigms in Network Management Applications, in *Proceedings of the 20th International Conference on Software Engineering (ICSE'98)*, April, Kyoto, Japan, 146-155.

Barr, W. J., T. Boyd, and Y. Inoue (1993). The TINA Initiative, *IEEE Communications Magazine*, March, 70-76.

Bates, J. (1994). The Role of Emotion in Believable Characters, *Communications of the ACM*, **37**(7):122-125.

Beni, G. and J. Wang (1989). Swarm Intelligence in Cellular Robotic Systems, in *Proceedings of the NATO Advanced Workshop on Robots and Biological Systems,* Italy.

Bieszczad, A. and B. Pagurek (1998). Network Management Application-Oriented Taxonomy of Mobile Code, in *Proceedings of the EEE/IFIP Network Operations and Management Symposium (NOMS'98)*, February 15-20, New Orleans, Louisiana, 659-669.

Bieszczad, A., S. K. Raza, B. Pagurek, and T. White (1998a). Agent-Based Schemes for Plug-And-Play Network Components, in *Proceedings of the Third International Workshop on Agents in Telecommunications Applications (IATA'98)*, July 4-7, Paris, France, 89-101.

Bieszczad, A., T. White, and B. Pagurek (1998b). Mobile Agents for Network Management, *IEEE Communications Survey*, **1**(1):http://www.comsoc.org/pubs/surveys/4q98issue/bies.html.

Bonabeau, E., M. Dorigo, and G. Theraulaz (1999). *Swarm Intelligence: From Natural to Artificial System*, New York: Oxford University Press.

Bonabeau, E., F. Henaux, S. Guerin, D. Snyers, P. Kuntz, and G. Theraulaz (1998). Routing in Telecommunication Networks with "Smart" Ant-Like Agents, in *Proceedings of the Second International Workshop on Intelligent Agents for Telecommunication Applications (IATA'98)*, July 4-7, Paris, France, 60-71.

Busuoic, M. and D. Griffiths (1993). Cooperating Intelligent Agents for Service Management in Communications Networks, in *Proceedings of the 1993 Workshop on Cooperating Knowledge Based Systems (CKBS'93)*, September, University of Keele, UK, 213-226.

Cheng, D. T. and S. Covaci (1997). The OMG Mobile Agent Facility: A Submission, in *Proceedings of the First International Workshop on Mobile Agents (MA'97)*, April 7-8, Berlin, Germany, 98-110.

Chess, D., B. Grosof, C. Harrison, D. Levine, and C Parris (1995). Itinerant Agents for Mobile Computing, *IEEE Personal Communications*, **2**(5):34-49.

Chin, D. (1991). Intelligent Interfaces as Agents, in J. Sullivan and S. Tyler (eds.), *Intelligent User Interfaces*, New York, NY: ACM Press, 177-206.

Csurgay, P., A. Oesleboe, and F. A. Aagesen (1997). Teleservices and Internet Application Technology, in *Proceedings of the IFIP TC6 International Symposium on Network Information Processing Systems*, October 14-16, Sofia, Bulgaria.

Dent, L., J. Boticario, J. McDermott, T. Mitchell, and D. Zabowski (1992). A Personal Learning Apprentice, in *Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI'92)*, July 12-16, San Jose, California, 96-103.

Eaton, P. S., E. C. Freuder, and R. J. Wallace (1998). Constraints and Agents: Confronting Ignorance, *AI Magazine*, **19**(2):51-65.

El-Darieby, M. and A. Bieszczad (1999). Intelligent Mobile Agents: Towards Network Fault Management Automation, in *Proceedings of the 6th IFIP/IEEE Integrated Symposium on Integrated Network Management (IM'99)*, May 24-28, Boston, Massachusetts, 611-622.

Esfandiari, B., G. Deflandre, J. Quinqueton, and C. Dony (1996). Agent-Oriented Techniques for Network Supervision, *Annals of Telecommunications*, **51**(9/10):521-529.

Etzioni, O., H. M. Levy, R. B. Segal, and C. A. Thekkath (1995). The Softbot Approach to OS Interfaces, *IEEE Software*, **12**(4):42-51.

Etzioni, O. and D. Weld (1994). A Softbot-Based Interface to the Internet, *Communications of the ACM*, **37**(7):72-76.

Ferguson, D. F., C. Nikolaou, J. Sairamesh, and Y. Yemini (1996). Economic Models for Allocating Resources in Computer Systems, in *Market-Based Control: A Paradigm for Distributed Resource Allocations*, S. Clearwater (ed.), New Jersey: World Scientific Press, 156-183.

Frei, C. and B. Faltings (1998). A Dynamic Hierarchy of Intelligent Agents for Network Management, in *Proceedings of the Second International Workshop on Intelligent Agents for Telecom Applications (IATA'98)*, July 4-7, Paris, France, 9-16.

Fuggetta, A., G. P. Picco, and G. Vigna (1998). Understanding Code Mobility, *IEEE Transactions on Software Engineering*, **24**(5):342-361.

Galliers, J. R. (1988). A Strategic Framework for Multi-Agent Cooperative Dialogue, in *Proceedings of the Eighth European Conference on Artificial Intelligence (ECAI'88)*, August, Munich, Germany, 415-420.

Gambardella, L. M. and M. Dorigo (1995). Ant-Q: A Reinforcement Learning Approach to the Traveling Salesman Problem, in *Proceedings of the Twelfth International Conference on Machine Learning (ML'95)*, July, Tahoe City, California, 252-260.

Garijo, F. J. and D. Hoffmann (1992). MAITE: An Operator Assistance Expert System for Troubleshooting Telecommunications Networks, in *Proceedings of the 3rd International Conference of Database and Expert Systems Applications (DEXA'92)*, Valencia, Spain, 14-19.

Meyer, K., M. Erlinger, J. Betzer, C. Sunshine, G. Goldszmith, and Y. Yemini (1995). Decentralizing Control and Intelligence in Network Management, in *Proceedings of the Fourth International Symposium on Integrated Network Management*, Santa Barbara, California, 4-16.

Gray, R. (1996). Agent-Tcl: A Flexible and Secure Mobile Agent System, in *Proceedings of the 4th Annual Tcl/Tk Workshop*, July, Monterey, California, 9-23.

Griffeth, N. D. and H. Velthuijsen (1993). Win/Win Negotiation Among Autonomous Agents, in *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence*, Hidden Valley, PA, 187-202.

Hylton, J., K. Manheimer, F. Drake, B. Warsaw, R. Masse, and G. van Rossum (1996). Knowbot Programming: System Support for Mobile Agents, in *Proceedings of the Fifth IEEE International Workshop on Object Orientation in Operating Systems (IWOOOS'96)*, October 27-28, Seattle, Washington, 8-13.

Hylton, J. and G. van Rossum (1997). Using the Knowbot Operating Environment in a Wide-Area Network, in *the 3rd ECOOP Workshop on Mobile Object Systems (MOS'97)*, June 9-10, Jyväskylä, Finland.

Jennings, N. R., K. Sycara, and M. Wooldridge (1998). A Roadmap of Agent Research and Development, *Autonomous Agents and Multi-Agent Systems*, **1**:275-306.

Joshi, N. and V. C. Ramesh (1998). On Mobile Agent Architectures, *Technical Report*, ECE Department, Illinois Institute of Technology.

Kraus, S., E. Ephrati, and D. Lehmann (1991). Negotiation in a Non-Cooperative Environment, *Journal of Experimental and Theoretical Artificial Intelligence*, **3**(4):255-282.

Kraus, S., J. Wilkenfeld, and G. Zlotkin (1995). Multiagent Negotiation Under Time Constraints, *Artificial Intelligence*, **75**(2):297-345.

King, J. A. (1995). Intelligent Agents: Bringing Good Things to Life, *AI Expert*, **10**(2):17-19.

Laasri, B., H. Laasri, S. Lander, and V. Lesser (1992). A Generic Model for Intelligent Negotiating Agents, *International Journal on Intelligent Cooperative Information Systems*, **1**(2):291-317.

Lashkari, Y., M. Metral, and P. Maes (1994). Collaborative Interface Agents, in *Proceedings of the Twelfth National Conference on Artificial Intelligence*, August 1-4, Seattle, Washington, 444-449.

Lesser, V. R. (1995). Multiagent Systems: An Emerging Subdiscipline of AI, *ACM Computing Surveys*, **27**(3):340-342.

Lesser, V. R., J. Pavlin, and E. H. Durfee (1988). Approximate Processing in Real Time Problem Solving, *AI Magazine*, **9**(1):49-61.

Lieberman, H. (1995). Letizia: An Agent that Assists Web Browsing, in *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI'95)*, August 20-25, Montreal, Quebec, Canada, 924-929.

Lingnau, A., O. Drobnik, and P. Dömel (1995). An HTTP-based Infrastructure for Mobile Agents, in *Proceedings of the 4th International WWW Conference*, December 11-14, Boston, Massachusetts, 461-471.

Maes, P. (1994). Agents that Reduce Work and Information Overload, *Communications of ACM*, **37**(7):31-40.

Maes, P. and R. Kozierok (1993). Learning Interface Agents, in *Proceedings of the Eleventh National Conference on Artificial Intelligence*, August 18-20, Washington, D.C., 459-465.

Magedanz, T., K. Rothermel, and S. Krause (1996). Intelligent Agents: An Emerging Technology for Next Generation Telecommunications?, in *Proceedings of INFOCOMM'96*, March 24-28, San Francisco, California, 464-472.

Martin-Flatin, J.-P. and S. Znaty (1997). A Simple Typology of Distributed Network Management Paradigms, in *Proceedings of the 8th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM'97)*, October 21-23, Sydney, Australia, 13-24.

Nwana, H. S. (1996). Software Agents: An Overview, *Knowledge Engineering Review*, **11**(3):1-40.

Osborne, M. J. and A. Rubinstein (1994). *A Course in Game Theory*, Boston: MIT Press.

Computational Intelligence in Telecommunications Networks, W. Pedrycz and A.V. Vasilakos (Eds.), CRC Press, 2000

Pagurek, B., Y. Li, A. Bieszczad, and G. Susilo (1998). Network Configuration Management in Heterogeneous ATM Environments, in *Intelligent Agents for Telecommunications Applications*, S. Albayrak and F. J. Garijo (eds.), Berlin: Springer-Verlag, 72-88.

Papazoglou, M. P., S. C. Laufman, and T. K. Sellis (1992). An Organizational Framework for Cooperating Intelligent Information Systems, *Journal of Intelligent and Cooperative Information Systems*, **1**(1):169-202.

Peine, H. (1997). Ara – Agents for Remote Action, in *Mobile Agents: Explanations and Examples*, W. Cockayne and M. Zyda (eds.), Greenwich, Connecticut: Manning, 96-163.

Peine, H. and T. Stolpmann (1997). The Architecture of the Ara Platform for Mobile Agents, in *Proceedings of the First International Workshop on Mobile Agents (MA'97)*, April 7-8, Berlin, Germany, 50-61.

Plu, M. (1998). Software Technologies for Building Agent Based Systems in Telecommunications Networks, in *Agent Technology: Foundations, Applications, and Markets*, N. R. Jennings and M. J. Wooldridge (eds.), Berlin, Germany: Springer-Verlag, 241-266.

Rao, A. S. and M. P. Georgeff (1995). BDI Agents: From Theory to Practice, in *Proceedings of the first International Conference on Multi-Agent Systems (ICMAS'95)*, June 12-14, San Francisco, California, 312-319.

Rosenschein, J. S. and M. R. Genesereth (1985). Deals among Rational Agents, in *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, Los Angeles, CA, 91-99.

Rosenschein, J. S. and G. Zlotkin (1994). *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers*, Boston, MA: MIT Press.

Sahai, A. and C. Morin (1998). Enabling a Mobile Network Manager through Mobile Agents, in *Proceedings of the Second International Workshop on Mobile Agents (MA'98)*, September, Stuttgart, Germany, 249-260.

Sahai, A., C. Morin, and S. Billiart (1997). Intelligent Agents for a Mobile Network Manager (MNM), in *Proceedings of the IFIP/IEEE International Conference on Intelligent Networks and Intelligence in Networks (2IN'97)*, September 2-5, Paris, France, 449-463.

Sandholm, T. and V. Lesser (1995). Coalition Formation among Bounded Rational Agents, in *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montreal, Canada, 662-669.

Schoonderwoerd, R., O. Holland, J. Bruten, and L. Rothkrantz (1996). Ant-Based Load Balancing in Telecommunications Networks, *Journal of Adaptive Behavior*, **5**(2):169-207.

Schoonderwoerd, R., O. Holland, and J. Bruten (1997). Ant-Like Agents for Load Balancing in Telecommunications Networks, in *Proceedings of the First International Conference on Autonomous Agents (Agents'97)*, February 5-8, Marina del Rey, California, 209-216.

Sheth, B. and P. Maes (1993). Evolving Agents for Personalized Information Filtering, in *Proceedings of the Ninth Conference on Artificial Intelligence for Applications*, Orlando, Florida, IEEE Computer Society Press, 345-352.

Shoham, Y. (1993). Agent-Oriented Programming, *Artificial Intelligence*, **60**(1):51-92.

Sycara, K. (1990a). Persuasive Argumentation in Negotiation, *Theory and Decision*, **28**(3):203-242.

Sycara, K. (1990b). Negotiation Planning: An AI Approach, *European Journal of Operational Research*, **46**(2):216-234.

Computational Intelligence in Telecommunications Networks, W. Pedrycz and A.V. Vasilakos (Eds.), CRC Press, 2000

Straßer, M., J. Baumann, and F. Hohl (1996). Mole—A Java based Mobile Agent System, in *Proceedings of the Second ECOOP Workshop on Mobile Object Systems (ECOOP'96)*, July 8-9, University of Linz, Austria, 301-308.

Weihmayer, R. and M. Tan (1992). Modeling Co-Operative Agents for Customer Network Control Using Planning and Agent-Oriented Programming, in *Proceedings of the IEEE Global Telecommunications Conference (Globecom'92)*, December 6-9, Orlando, FL, 537-543.

Weihmayer, R. and H. Velthuijsen (1998). Intelligent Agents in Telecommunications, in *Agent Technology: Foundations, Applications, and Markets*, N. R. Jennings and M. J. Wooldridge (eds.), Berlin, Germany: Springer-Verlag, 241-266.

White, J. E. (1996). Mobile Agents, in *Software Agents*, J. Bradshaw (ed.), AAAI Press/MIT Press, 437-472.

White, T., A. Bieszczad, and B. Pagurek (1998a). Distributed Fault Location in Networks Using Mobile Agents, in *Intelligent Agents for Telecommunications Applications*, S. Albayrak and F. J. Garijo (eds.), Berlin: Springer-Verlag, 130-141.

White, T. and B. Pagurek (1998). Towards Multi-Swarm Problem Solving in Networks, in *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98)*, July 2-8, Paris, France, 333-340.

White, T., B. Pagurek, and F. Oppacher (1998b). Connection Management using Adaptive Agents, in *Proceedings of the 1998 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'98)*, July 13-16, Las Vegas, Nevada, 802-809.

White, T., B. Pagurek, and F. Oppacher (1998c). ASGA: Improving the Ant System by Integration with Genetic Algorithms, in *Proceedings of the 1998 Conference on Genetic Programming (GP'98)*, July 22-25, Madison, Wisconsin, 610-617.

Woodridge, M. and N. Jennings (1995). Intelligent Agents: Theory and Practice, *The Knowledge Engineering Review*, **10**(2):114-152.

Yemini, Y. (1993). The OSI Network Management Model, *IEEE Communications Magazine*, **31**(5):20-29.

Yemini, Y. and S. da Silva (1996). Towards Programmable Networks, in *Proceedings of the 1996 IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM'96)*, October 28-30, L'Aquila, Italy.

Zeng, D. and K. Sycara (1996). Bayesian Learning in Negotiation, in *Working Notes of the AAAI Spring Symposium on Adaptation, Co-Evolution and Learning in Multiagent Systems*, Stanford, CA.

Exercise Problems:

1.   Why do today's telecommunication networks require better handling of data?

2.   What has motivated the multiagent systems (MAS) in telecommunication networks? What are the advantages of agent technologies and intelligenec in telecommunication networks?

3.   What areas in telecommunication networks can intelligent agents be of help and why?

4.   What are some of the attributes sufficient or necessary for defining an agent (list at least 6)?

5.   What makes an agent *intelligent*?

6.   List some advantages of a multiagent system over a centralized system.

7.   Describe four architectures of management systems for telecommunication networks and identify the differences.

8.   Describe some mobile agents and their applications in telecommunication networks.

9.   How can intelligent interface agents be useful in telecommunication networks?

10.   What is swarm intelligence? What is an ant system? How can an ant system be useful in telecommunication networks?

11.   What is a lieu in the environment of a Mobile Network Manager (Sahai *et al.* 1997)?

12.   How is service management and provisioning related to telecommunications?