EFFICIENT MANAGEMENT AND CONTROL OF TELEMETRY RESOURCES

Alan E. Cowart, Ph.D.
Scientific Research Corporation

Michelle Baldonado
AFFTC/ARTM

ABSTRACT

In recent years the telemetry community has encountered a growing demand for bandwidth from users and a corresponding loss of spectrum. The Advanced Range Telemetry (ARTM) Program has responded to this situation with an initiative to develop, demonstrate, and improve the management and control of telemetry resources using demand assigned multiple access (DAMA) techniques. This initiative has proceeded along two paths. The first path is in the development of an expert system to facilitate the scheduling of telemetry missions and the deconfliction of their frequencies. This system emphasizes the graphical manipulation of mission data and uses a genetic algorithm to search for an optimal set of mission frequencies. The second path is the development of a bidirectional command and control link to remotely control and configure the frequency of a telemetry link. This link uses the simple network management protocol (SNMP) over a wireless Internet Protocol (IP) network implemented with Digital Communications Network System (DCNS) units.

KEY WORDS

ARTM, scheduling, frequency deconfliction, expert system, genetic algorithm, optimization, remote configuration, command and control link, SNMP, IP

INTRODUCTION

In recent years the telemetry community has encountered a growing demand for bandwidth from users and a corresponding loss of available spectrum. This situation led to the establishment of the Advanced Range Telemetry (ARTM) Program and their support of innovative technologies to improve the utilization of the telemetry spectrum. One aspect of this overall problem is the manner in which telemetry resources (in particular frequency) are managed on a daily basis. The process of deconflicting

frequency requests may be largely unknown to users but the difficulties it presents is all too familiar to schedulers and frequency managers. Many of the techniques as practiced today are manually intensive and inefficient and have remained relatively unchanged as scheduling and frequency deconfliction tasks have increased in complexity. Only recently have computerized and automated processes been considered and embraced to any degree for this task.

This paper describes a computer-based approach to the problem of deconflicting frequencies and managing other telemetry resources that was developed as part of a Small Business Innovative Research (SBIR) initiative from ARTM. This system has been named Scheduling Management for Advanced Range Telemetry (SMART) and incorporates computer and network-based solutions to alleviate the load on schedulers and frequency managers and improve their efficiency.

SMART OVERVIEW

The SMART system concept evolved as part of an examination of the current frequency deconfliction and range scheduling processes, their interaction, and the challenges they faced. Several key objectives of the SMART system emerged during this period and guided its development:

1. The system should be accessible by a wide range of personnel – users, schedulers, frequency managers, and area frequency coordinators. The actual process that SMART is intended to support begins with mission requests for range assets (of which frequency is only one) and eventually involves inter-range frequency contention and coordination.

2. Mission data should be graphically displayed and manipulated as much as possible to facilitate ease of use. Graphical user interfaces (GUIs) have become a standard part of computer applications and are in part, responsible for the proliferation of information technology.

3. The system should enable schedulers, frequency managers, and area frequency coordinators to deconflict frequency requests quicker and easier. Furthermore, it should lead to more efficient use of the available spectrum. This objective leads to recasting the frequency deconfliction process as an optimization problem.

4. The system should be able to remotely configure and monitor telemetry assets, particularly the frequency and configuration of transmitters, receivers, and other link equipment. This objective advances the notion of a telemetry command and control link and is further inspired by the development and implementation of network management tools.

5. The system should utilize off-the-shelf components as much as possible. This is a well-known anthem in both the government and commercial sectors and is intended to prevent the continual re-invention of the wheel.

Figure 1 is a functional block diagram of the resulting SMART system. The system implements two major functions: resource scheduling, and remote configuration. Portions of the system have been implemented and its continued evolution and upgrade is facilitated by its choice of components.

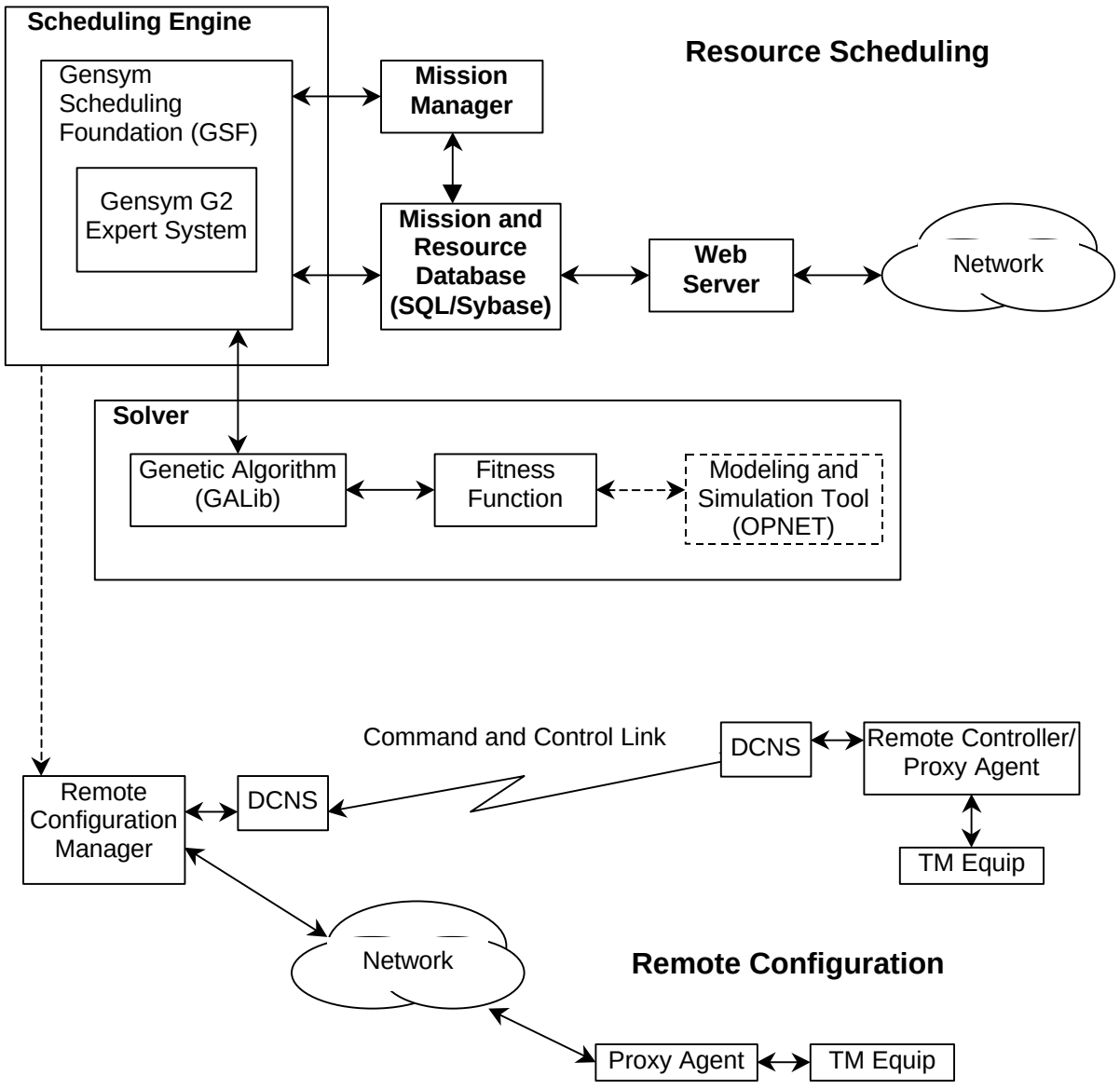

Figure 1. SMART Functional Block Diagram.

RESOURCE SCHEDULING

The resource scheduling portion of SMART is implemented on a Windows NT workstation and consists of five major components: (1) Scheduling Engine, (2) Mission and Resource Database, (3) Mission Manager, (4) Solver, and (5) Web Server.

Scheduling Engine

The Scheduling Engine consists of a Gensym G2 Expert System with the Gensym Scheduling Foundation (GSF). There are over 10,000 G2 licenses worldwide and GSF has been adapted for a variety of process control applications ranging from steel rolling mills to a brewery. G2 also runs on both UNIX and Windows 95/98/NT platforms, and its knowledge base is compatible between these different platforms. From the Scheduling Engine a SMART user can read, write, and display a daily schedule, graphically manipulate mission times, access the Solver for frequency deconfliction, and access the Mission Manager to manipulate all mission parameters. The functions of the Scheduling Engine are primarily implemented with pull-down menus and click-and-drag mouse operations, making it relatively user-friendly.

Mission and Resource Database

The Mission and Resource Database is a SQL database for maintaining information on missions and the resources that support them. In the context used here, resource data refers to range and program assets and their characteristics. Mission data refers to the collection of resources that are utilized in a mission, the specific configuration of those resources, and other mission-specific data and identifiers. For instance, aircraft, transmitters, receivers, antennas, test areas, and receiver sites are all resources. The database must also keep track of the configuration options and specifications for the resources. These include such things as the modulation types, data rates, and filter choices for receivers and the coordinates of the sites where the receivers are located. A typical mission requires a telemetry link consisting of a transmitter and antenna on an aircraft flying in a test area during a certain period of time and being received at a receiver site through an antenna and receiver tuned to the frequency authorized for the mission. As anyone who has seen this array of equipment can testify, there are a multitude of knobs, switches, dials, and other parameters to set if the link is to operate correctly. The mission data specifies the selection of resources required for a mission, their configuration, and any link performance requirements. Link performance can be specified in minimum signal-to-noise ratio or maximum bit error rate. Although performance is not typically specified today, it is an example of a service that SMART will enable.

Mission Manager

The Mission Manager is a graphical interface to the Mission and Resource Database. This component enables a SMART user to access and manipulate all mission and resource data. It is available as a stand alone application or from within the Scheduling Engine.

Solver

The Solver is an extension of the Scheduling Engine that is used to identify frequency conflicts in schedules and determine sets of frequencies for missions to operate on. The frequencies associated with different telemetry links conflict if their mutual interference exceeds certain levels. There are several ways to identify a frequency conflict: (1) frequency overlap, (2) signal-to-interference ratio, and (3) link performance modeling and simulation.

The first technique is widely used today and has been implemented in the Solver. Link bandwidth is specified such that the sidelobe levels at the band edges are sufficiently suppressed in all conceivable scenarios that they will not adversely interfere with adjacent links. Any overlap between the bandwidths of adjacent links indicates a frequency conflict. The greatest advantage of this technique is its simplicity. However, it tends to lead to inefficient frequency sets since it does not account for important link parameters such as propagation loss, antenna patterns, the relative orientation of link paths, and transmit power levels. These factors may be considered on an individual basis if a cursory view of the distribution of links in a region indicates that there is a possibility that a conflict does not actually exist.

The second technique computes the interference between all links operating in a region and uses the signal-to-interference ratio to determine if there is a frequency conflict. This technique is in the process of being implemented in the Solver. It requires the full range of information on missions and resources that resides in the database. It also requires that a minimum signal-to-interference ratio be specified for a link to differentiate between acceptable and unacceptable link performance. With this technique unacceptable performance indicates a frequency conflict.

The third technique requires the integration of a modeling and simulation tool into the Solver to perform the link performance calculations. The OPNET Modeler/Radio is a commercially available tool from MIL3 that has been identified for future implementation in the Solver. This is a discrete event simulation tool that is capable of performing detailed analyses of communication, network, and computer systems. The performance statistics that can be generated include signal-to-noise ratios and bit error rates as a function of time. It also has well-defined application program interfaces (APIs) to facilitate its integration into applications such as SMART. While this approach can

generate very detailed performance estimates, it does have the disadvantage of requiring moderate execution times.

Determining a set of frequencies for the missions in a schedule to operate on can be viewed as an optimization problem:

Given:	Set of missions, the L links that must be supported, and their characteristics
Find:	Set of frequencies $F = \{f_i, i=1, 2, \ldots L\}$
Such that:	Mutual interference is minimized

The characteristics of a mission and the links it requires can be a fairly extensive list that includes mission identifiers, start and stop times, equipment options, and site coordinates. One approach to solving an optimization problem of this sort is to use a genetic algorithm. Genetic algorithms are a class of robust search and optimization algorithms based on the principles of natural selection and genetics [1]. Parameter values to be optimized are encoded into sets of strings in which the parameters are referred to as genes and the strings as chromosomes. In this case, each frequency request is mapped into a gene and a set of frequencies to support all of the missions in a schedule becomes a chromosome. The value of a gene reflects the frequencies that the mission associated with the gene can use while a chromosome describes a possible set of operating frequencies for the missions.

The genetic algorithm manipulates a set of chromosomes using three operators based on probabilistic transition rules: 1) reproduction, 2) crossover, and 3) mutation. Reproduction is based on the evaluation of a fitness function for each chromosome. Chromosomes receiving higher fitness values have a higher probability of being reproduced or copied into succeeding generations. This is the “survival of the fittest” principle. Crossover involves the exchange or swapping of genes between chromosomes. Mutation involves random changes in individual genes. The set of chromosomes generated at each stage in the genetic algorithm is a generation. Each generation is subjected to reproduction, crossover, and mutation until an acceptable solution results or it is determined that a solution cannot be found.

There are two critical tasks to be accomplished in applying a genetic algorithm to a problem: encoding parameters into chromosomes, and determining a fitness function. In this case, each gene is associated with a link and the value of a gene points to the list of frequencies that the link is authorized or able to use. These frequencies can be represented as discrete frequencies or ranges of frequencies. Link frequencies can also be reserved, indicating that the frequency has been set aside for that link only.

The value of the fitness function must be non-negative, and higher fitness values must reflect better parameter sets. A two-stage function was devised for SMART. The first stage is a feasibility function and the second stage is an interference metric:

$G_{fitness}(F) = G_{feasibility}(F) + u(G_{feasibility}(F)-1)\, G_{interference}(F)$ where: $u(x)$ is the unit step function.

The feasibility function indicates whether the minimum requirements of all links are satisfied. A value of zero indicates that none of the minimum link requirements are satisfied for the duration of a schedule while a value of 1 indicates that all of the requirements are satisfied. The interference metric indicates the extent to which the minimum requirements are exceeded and is utilized only if a frequency set is feasible (i.e., $G_{feasibility}(F)=1$). The genetic algorithm continues to execute until it is determined that a feasible solution cannot be found or until the difference between fitness values between generations does not change significantly. The chromosome associated with the highest fitness value indicates the best solution.

Web Server

The Web Server enables users to access the database via a network and a web browser. This capability is similar to web access that is available to the Mojave Coordination Group.

REMOTE CONFIGURATION

The remote configuration portion of SMART is a straightforward adaptation of network management technology to the telemetry environment. This capability was deemed to be an important part of the overall SMART concept because the resource scheduling function takes a big step toward relieving telemetry and frequency management personnel of the burden of having to manually select and deconflict mission frequencies. The remote configuration function is an extension of this concept to ensure that links are setup quicker and more accurately. The simple network management protocol (SNMP) was chosen for implementation since it is the most widely used approach to network management and is defined in a series of Internet Engineering Task Force (IETF) Request for Comments (RFCs). There are numerous products available to support it. A telemetry command and control link would also be needed to set the frequency of an airborne telemetry transmitter. The ARTM chose the Digital Communications Network System (DCNS) developed by L3 Communications for a prototype of this link. A Microdyne 700MR receiver and Miteq 9400 Upconverter were selected as telemetry equipment to be remotely configured.

An SNMP-based managed network contains several elements [2]:

1. One or more managed nodes, each containing a processing entity called an agent or associated with a proxy agent.
2. One or more management stations, each containing a processing entity called a manager.
3. Entities which can perform both manager and agent roles (optional).
4. Management information in each managed node which describes the node configuration, state, and statistics and which controls the actions of the node.
5. A management protocol used to exchange messages.

As it is used here, a managed node refers to any telemetry equipment that can be configured remotely. In a typical network, a managed node has an assigned network address and agent software running on it. However, this is not always possible. Older legacy systems may not support network standards, small systems may be unable to run agent software, and other systems may support different standards for remote configuration. In these cases, a proxy agent acts on behalf of a node [3]. Telemetry equipment and its configuration does not typically support network operation yet. As telemetry systems begin to embrace network architectures, users and vendors may wish to consider if equipment with network interfaces is desirable.

The SNMP management information on a managed node is specified in a Management Information Base (MIB). A MIB defines the objects to be monitored, configured, and controlled in a network. The structure of a MIB also includes a scheme to uniquely identify objects. Published MIBs include management information on the characteristics of systems, interfaces, and various protocols such as IP, TCP, UDP, ICMP, and EGP. Other MIBs specific to equipment at a node cover the functions, events, and statistics of the equipment. Again, as telemetry systems begin to adopt network characteristics, users and vendors may want to consider whether to supply standard MIBs for the control and monitoring of their equipment.

The SNMP protocol runs over the User Datagram Protocol (UDP) and IP stacks found on most PC and UNIX platforms. This approach facilitates the adaptation of the command and control link to a variety of data links. A different data link can be accommodated by encapsulating the IP packet within the payload of the new data link protocol.

CONCLUSION

This paper has described a system for scheduling missions, maintaining mission and resource data, deconflicting frequencies, and remotely configuring the resources required for a mission. Portions of the system are in the process of being implemented and tested. Unique features of this system are a technique for automating and optimizing the

deconfliction of frequencies and the use of a command and control link to remotely configure telemetry equipment.

ACKNOWLEDGEMENTS

This work has been supported by ARTM as part of Air Force SBIR topic AF96-220 through Contracts F04611-97-C-0073 and F04611-96-C-0062. The authors wish to thank Mr. Chuck Irving of ARTM for his support, encouragement, and understanding in the pursuit of this work.

REFERENCES

[1] Goldberg, David E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.

[2] Perkins, David and McGinnis, Evan, Understanding SNMP MIBs, Prentice Hall, Upper Saddle River, NJ, 1997.

[3] Stallings, William, SNMP, SNMPv2, and CMIP: The Practical Guide to Network Management, Addison-Wesley, Reading, MA, 1993.