ORIGINAL PAGE IS
OF POOR QUALITY

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO.	2.	3. Recipient's Accession No
4. Title and Subtitle ENGINEERING CALCULATIONS FOR SOLVING THE ORBITAL ALLOTMENT PROBLEM			5. Report Date August 1988
			6.
7. Author(s) C. Reilly, E.K. Walton, C. Mount-Campbell, R. Caldecott, E. Aebker, F. Mata			8. Performing Organization Rept. No 718688-7
9. Performing Organization Name and Address THE OHIO STATE UNIVERSITY ELECTROSCIENCE LABORATORY 1320 Kinnear Road Columbus, Ohio 43212			10. Project/Task/Work Unit No
			11. Contract(C) or Grant(G) No (C) (G) NAG3-159
12. Sponsoring Organization Name and Address NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Lewis Research Center 21000 Brookpark Rd. Cleveland, Ohio 44135			13. Type of Report & Period Covered Technical
			14.
15. Supplementary Notes			

16. Abstract (Limit: 200 words)

In this report, four approaches for calculating downlink interferences for shaped-beam antennas are described. An investigation of alternative mixed-integer programming models for satellite synthesis is summarized. Plans for coordinating the various programs developed under this grant are outlined. Two procedures for ordering satellites to initialize the k-permutation algorithm are proposed. Results are presented for the k-permutation algorithms. Feasible solutions are found for five of the six problems considered. Finally, it is demonstrated that the k-permutation algorithm can be used to solve arc allotment problems.

17. Document Analysis a. Descriptors

Satellite synthesis
Shaped-beam antennas
Allotment planning
Integer Programming

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report) Unclassified	21. No of Pages 29
	20. Security Class (This Page) Unclassified	22. Price

(See ANSI-Z39.18) See Instructions on Reverse

OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce

Contents

PRECEDING PAGE BLANK NOT FILMED

List of Tables

List of Figures

Chapter 1

Purpose

The purpose of this grant is to develop methods and procedures, including computer codes, for performing engineering calculations that will be useful for the United States delegations to international administrative conferences concerning satellite communications. Our attention has been directed exclusively toward Fixed Satellite Service (FSS) issues during the interim 15 January 1987 to 14 January 1988, since this service will be a major topic at the World Administrative Radio Conference to be held later this year (WARC-88).

Chapter 2

Shaped-Beam Antenna Studies

2.1 Introduction

Up to this time, the required satellite separation calculations have been made under the assumption that satellite antennas are transmitting and receiving elliptical gain patterns. This means that the loci of constant directivity in a plane normal to the boresight of the antenna were assumed to be elliptical. The -3 dB contour of the antenna pattern was defined by the smallest ellipse which would enclose the service area. The service area was defined by a set of test points designed to represent it.

In actual practice, it is common for satellites to be designed with shaped beams. In this case, the beams of the satellite transmitting and receiving antennas are shaped to more closely follow the contour of the service area. One common technique for generating such beams is to use smaller multiple spot beams which cover the service area. These spot beams can be generated by a set of multiple feeds. Although such techniques have been in use for some time, no standardized method of calculating the resultant interference from such antennas to regions outside their service areas has yet been adopted by the CCIR. As part of this research project, a number of possible representations of the shaped beam patterns were considered and the effect of each on satellite spacings was illustrated. The techniques are described in detail in the October 1987 Technical Report by Kohnhorst, Levis and Walton [1].

The techniques are outlined in the next section.

2.2 Modeling of Shaped Beam Antenna Patterns

The models for future shaped beam antenna developments that are described in the technical report mentioned above are described briefly below.

1. *Elliptical Half-Power Patterns*

   This method is based on a reference pattern published in the Final Acts of RARC-83 [Vol. 20, p. 150]. The pattern is elliptical and the relative gain is a function of the off-axis angle. The pattern function is similar to the previous single beam pattern up to the location of the -3 dB ellipse, but beyond this ellipse, the pattern function rolls off much more rapidly.

2. *Service Area Polygons*

   The next three methods consider the service area to be projected onto a plane normal to the boresight between the satellite and the aimpoint. Small distortions between the spherical earth and the projection onto this plane are ignored. Polygons are generated on this plane by projecting the service area testpoints onto it.

   - *Projections Through Polygon Center*

     This procedure first requires that the polygon formed on the projection plane be converted into a convex shape. This is done by starting at any projected test point and moving around the surface with projected lines which skip over any convex parts of the polygon. (See pg. 57 of Kohnhorst.)

Now, for the satellite using multiple spot beams, the power will fall off more rapidly outside this contour because of the smaller half-power beamwidth of the spot beams. The model thus considers that the half power beamwidth is *not* proportional to the total distance across the polygon at a each point, but *is* proportional to some integer fraction of this distance that represents the reciprocal of the number of possible spot beams across the polygon at the particular location.

Note in this case that the roll-off outside the polygon would be more rapid for a narrow "cut" through the polygon than for a wide "cut". This is the pattern behavior expected from a typical multi-feed shaped beam antenna.

- *Uniform Roll-off From the Boundary*

  In this model, the polygon representing the service area is assumed to be covered with a set of circular sub-beams of uniform size. (This would occur with a circular reflector antenna with a number of feeds.) The -3 dB contour of each of the boundary sub-beams is taken to lie on the border of the convex polygon derived from the original polygon. For each point outside the polygon, the shortest distance to the polygon is found, and the power computed based on the distance from the nearest associated sub-beam (i.e., the sub-beam having a -3 dB contour on the border of the polygon).

- *Specific Distribution of Sub-Beams*

  In this method, the polygon (original or convex) is covered with a specific fixed distribution of uniform circular sub-beams. The particular distribution is hand chosen using the subjective judgment of the user. In practice, this may differ from any specific final design for a particular antenna coverage requirement, but the example is useful for a demonstration of the technique.

> Once the locations of the sub-beams are known, the gain contours outside the service area are determined using superpositon of idealized voltage patterns for each sub-beam. Two different sub-beam roll-off patterns are demonstrated in reference [1].

The gain contour plots are given in the October 1987 technical report [1] (section 3.6.). They show the trade-offs that result from these different representations of shaped beam patterns. As might be expected, they all show improvements in terms of a reduction in required satellite spacing.

Chapter 3

Alternative Mixed-Integer Programming Models for Satellite Synthesis

Much of the recent effort to solve satellite synthesis problems has been concentrated on the problem of minimizing the sum of the absolute deviations between prescribed and desired satellite locations. The choice of this objective function was made at the time the study of integer programming models for satellite synthesis was begun. Though this objective represents a reasonable selection, there are other reasonable choices. As a result, a study of alternative mixed-integer programming (MIP) models with different objective functions has been conducted. All of these models are similar in form and use the minimum required satellite separation concept developed by Wang et al. [2]. The models and the investigations undertaken with respect to these models have been presented in Bhasin and Reilly [3]. A brief summary of the more important findings of this investigation is given below.

Specifically, the following point allotment models have been studied:

- Minimizing the sum of the absolute deviations between allotted and desired satellite locations.

- Minimizing the sum of weighted absolute deviations between allotted and desired satellite locations, where the weight for each satellite is inversely proportional to the length of its service arc. (Service arc constraints are not explicitly enforced.)
- Minimizing the largest of the absolute deviations between a satellite's allotted and desired locations.
- Minimizing the distance between the easternmost and westernmost allotted satellite locations.
- Maximizing the smallest of the actual separations between satellites beyond the corresponding minimum required separation.
- Maximizing the smallest gap between adjacent satellites.

Model 1 is the primary model. The second model was selected so it could be determined if there is a computational advantage to using a weighted objective function instead of explicitly enforcing service arc constraints as is done in Model 1. Model 3 was chosen so that the magnitudes of the absolute deviations will be more nearly equal for all satellites than they will be with either of the first two models. The fourth model is for the same problem studied by Ito et al. [4]; however, an MIP model is used instead of a nonlinear programming model. The fifth model is similar, in terms of its purpose, to the earlier nonlinear programming synthesis model developed at Ohio State because it attempts to maximize single-entry carrier-to-interference (C/I) ratios. It is also expected to leave room between satellites that could be used to accommodate satellites that are deployed later. Finally, Model 6 is expected to yield solutions similar to those of Model 5, but the establishment of gaps between satellites is dealt with more directly.

Several scenarios with between 10 and 13 satellites were used in the investigation of these problems. Each model was solved for each scenario with an MIP package.

The solutions were evaluated on the basis of observed convergence and solution time. The robustness of these models was also investigated to determine if one model might be preferred over another because it can produce good solutions to more than one model at less computing expense.

Feasible solutions were found most quickly for Models 1, 2, and 3. These same models also produced more optimal solutions than the other models did in the limited-time runs made. Furthermore, feasible solutions to these models appear to possess the properties they were expected to have. For example, Models 5 and 6 yielded solutions in which the satellites were allotted locations that spread the satellites over the available arc segment. In the event that C/I maximization is selected as the objective, then either of these models might be selected as a means to approximately maximize C/I ratios.

It has also been observed that the solutions to Models 1, 2, and 3 are good solutions to Model 4 as well. This indicates that the primary integer programming model (Model 1) seems to yield good solutions to the integer programming analog of the nonlinear programming model of Ito et al. [4]. The converse is not true. Solutions to Model 4 do not appear to be particularly good when they are evaluated in Models 1, 2, and 3.

This phenomenon occurs, first of all, because Models 1, 2, 3, and 4 favor solutions in which satellites are positioned at longitudes over their service areas in test problems like the ones used in this study. Since the default for a satellite's desired location is the midpoint of its service arc, if satellites are ordered in such a way that they may be positioned almost directly over their service areas (Models 1, 2, and 3) and their service areas are near one another, then the orbital arc segment in which the satellites are positioned would tend to be relatively short (Model 4). But, if the satellites are ordered in such a way that they can occupy a short arc segment (Model 4), there is no reason to think that the satellites will be positioned near the midpoints of their service arcs (Models 1, 2, and 3).

It is clear from this investigation that one cannot rely on relatively straightforward MIP models to yield consistently good solutions to synthesis problems in a reasonable amount of computing time. This is especially true for synthesis problems with many satellites. The synthesis model recommended by Mount-Campbell et al. [5] appears to be a much better choice for finding solutions to large synthesis problems than any of the models considered in this investigation because of its exploitable mathematical structure and the fact that a special-purpose solution procedure, a k-permutation algorithm [3], that exploits this structure has been developed under this grant.

Despite the fact that the usefulness of the MIP formulations presented in Bhasin and Reilly [3] seem to be limited to small synthesis problems, this investigation has indicated that the objective function that has been selected and can be exploited with the k-permutation algorithm leads to synthesis solutions that are viewed favorably when they are evaluated in the model of Ito et al. [4], the most widely accepted synthesis model to date. Since the converse is not true, solutions to Model 1 may be preferred over solutions to the model by Ito et al. on the basis of higher angles of elevation.

Chapter 4

Software Coordination Program Package

4.1 Introduction

The program package, developed under this project, consists of four principal programs which are normally executed in sequence as shown in Figure 4.1. The output from one program is used as input by the next in order, generally with some additional information.

These programs were developed by different people; in some cases several individuals contributed to a single program. As a result the data formats are not always compatible from one program to another; it is necessary to do some hand editing of the data between one program and the next. Also the formats used are in some cases very rigid; the hand editing must conform to an exact column format.

To make the programs more generally useful some modifications are required. To this end a number of changes have been initiated which will be completed and refined during the remainder of the project. The changes include the following:

- Change program syntax as needed to conform to standard Fortran 77.

- Modify the input and output formats of the programs so that data from one program that is needed by another is directly readable without hand editing of files.
- Further modify the programs and their input formats to make them as user-friendly as possible. This includes the removal, as far as possible, of fixed file names permanently built into programs, allowing the user freedom of choice in these. It also includes the introduction of the EASYCOM subroutine package to handle data input. This is designed to give great flexibility in the format of input files.

4.2 The EASYCOM Subroutines

The EASYCOM subroutines are a set of subroutines which have been developed at the ElectroScience Laboratory over a number of years. The routines were originally intended to facilitate the writing of user-friendly, interactive programs. However, they also accept input entirely from files so that the programs may be run in batch mode. They allow considerable freedom in the format of the data files including the addition of comments and selective skipping of portions of the data. The package contains routines of the form:

```
N=====CALL ECOMD('ABCD',NN)
```

where ABCD is a command mnemonic. (N and NN are statement numbers.) This allows execution of a program from a number of starting points in response to simple commands of the user's choice from the keyboard or a file.

```
======CALL ERQCHR('MESS',ITEM,NQUIT)
```

where MESS is a prompt message and ITEM is a character variable (routines are also available for the input of real and integer data). This routine issues the prompt message and requests input data from the keyboard or a file. (A complementary routine provides for data output.) The statement number NQUIT provides for changing the course of the program on command during data input.

The routines allow typing ahead. Multiple inputs may be typed on the same line separated by commas or spaces. No more prompts will be issued until the available data is exhausted. Help is available by typing a question mark. This will produce a list of available commands or the present value of the data ITEM. Input may be assigned to a file with a log kept on the terminal or another file if desired. Input files may include comments which are ignored by the program.

IF directives allow selective reading of input files. PAUSE and STOP directives allow keyboard input to be intermixed with file input.

When the program is run interactively, most input errors are trapped and appropriate messages issued without crashing the program.

4.3 Initial Modifications to the ELLARC and DELTA Programs

ELLARC and DELTA are the first two programs in the series of computer programs developed as part of this research project. ELLARC calculates the axes and orientation of the minimum-width elliptical beam which will cover a service area as viewed from various orbital locations. DELTA takes this beam data for a number of service areas and calculates the minimum separation for the satellites which serve these areas while maintaining a specified minimum carrier-to-interference ratio.

The input sequences for the ELLARC and DELTA programs are being modified to make them more user friendly. The output sequence of ELLARC is also being modified to make it compatible with the input to DELTA. The modified versions of both programs will use the EASYCOM input subroutine package and have been given the names ELLARCEC and DELTAEC to distinguish them from the previous versions.

The new versions will, by default, run from a command procedure file. In this event no keyboard input is needed. If that file is not found then a command file name will be requested from the keyboard. The name of the default command file is the only file name built into the program. Additional input and output file names that are needed appear in the command file rather than being built in. Thus these names may be changed without the necessity of recompiling the program.

ELLARCEC requires a file containing the latitudes and longitudes of the points defining each service area in addition to the command file. The original service area file used with ELLARC contained countries in various groupings, some countries appearing in more than one group. A new file was produced for use with the modified programs by sorting the service area names alphabetically and removing the duplicate entries.

Sample command and service area input files for ELLARCEC are shown in Figures 4.2 and 4.3. They illustrate the EASYCOM input format. Note the use of comments in these files. An exclamation point and everything following it on the same line is considered a comment. Note also the use of the IF directive ahead of each country name in the service area file. The data for each country (between one IF and the next) is only read if the name of that country appears in the command file. This removes the need for different service area files for each group of service areas for which ELLARC is run.

Modifications to the input of DELTAEC will be similar. The output from ELLARCEC (which will be read as input by DELTAEC) contains results which are not used by DELTAEC. The output format will be modified so that these appear as comments in the file. They will then be available for the human reader but will be ignored by DELTAEC. The output from ELLARCEC is listed by service areas. In the modified output each area name will be preceded by the IF directive so that the file may subsequently be read selectively by DELTAEC if desired; only those service areas appearing in the DELTAEC command file will be read.

4.4 Future Work

In the coming months modification of the ELLARCEC and DELTAEC programs will be completed and the other two programs modified as well. The modifications will include the construction of sample command files and other input data files not generated by programs in the series.

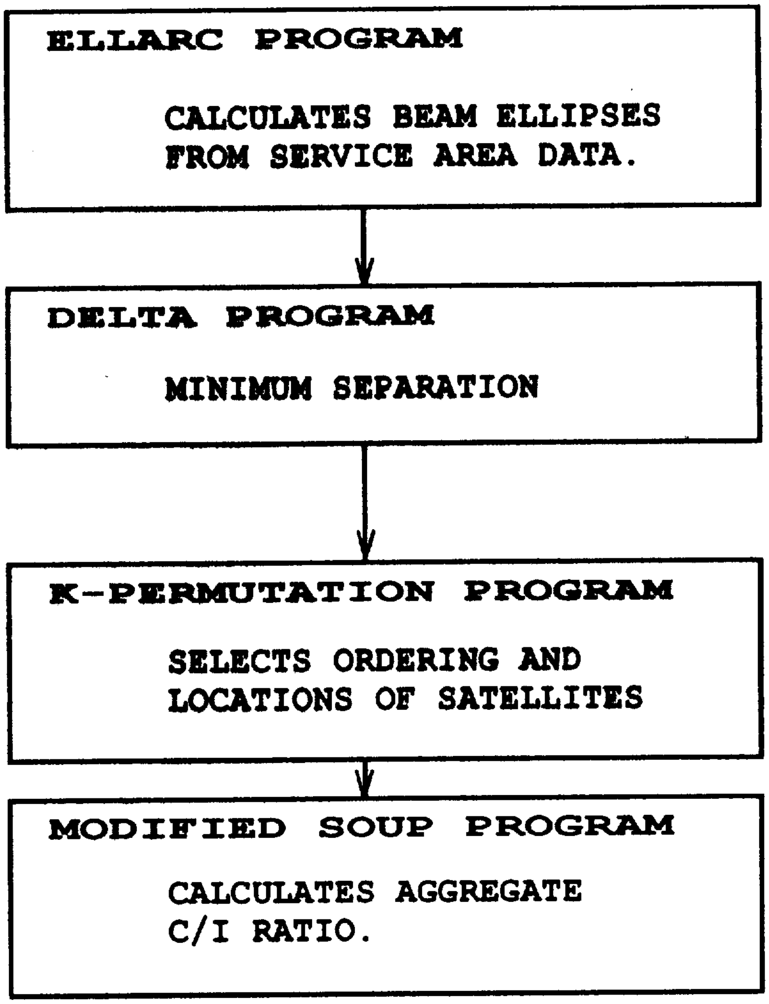

Figure 4.1: Sequence of program execution.

- File ELLARC.CF

  This is a command input file for the ELLARCEC program. Either of the following file names may be replaced with an EASYCOM PAUSE directive to request the names from the terminal at run time.

  - SERVAREAS.DAT Name of file containing service area data
  - ELLARCOUT.DAT Name of output file
  - 2 Number of service areas
  - 10.0 Minimum permitted elevation angle (deg)
  - 2.0 Spacing between calculated ellipses (deg)
  - 0.1 Satellite antenna pointing error
  - 1.0 Orientation angle error
  - 0.6 Minimum allowed width of elliptical beam

- The following are service area names. They should be separated by commas or appear on separate lines. They may be multiword names with embedded spaces (e.g., United States). Leading and trailing spaces are ignored. Up to the first 60 characters of the name will be used to locate data in the service area file. Names must appear in the same order as in the service area file though there may be fewer names in the following list. Only those names in this list will be used.

  - CAMBODIA
  - CHINA

Figure 4.2: Sample command file for the ELLARCEC program.

File SERVAREAS.DAT

```
   !This file is used as input by the program ELLARCEC.FOR. Each block will
!only be read if the country name in the EASYCOM IF test string matches that
!appearing after the IF on one of the lines in this file. Country names may be up
!to 60 characters in length and may have multiple words. Imbedded spaces must
!match exactly. The test string starts with the first character after the IF that is
!not a space or comma. It ends at a comma, exclamation point or the end of the
!line. Trailing spaces before the final comma or exclamation point are ignored.
   !The number on the line following the country name indicates the number of
!latitude and longitude pairs of the test points defining the perimeter of that coun-
!try. Subsequent lines contain the latitude and longitude pairs. The countries are
!listed alphabetically.
!
IF CAMBODIA
7 ! Number of points defining country
13.5 102.0 ! Latitude and longitude of point
14.5 104.0
14.8 106.7
13.0 107.8
10.8 106.2
10.1 104.5
12.0 102.5
IF CHILE
7
-56.00 -69.00
-46.00 -76.00
-44.00 -71.00
-34.00 -72.00
-23.00 -66.50
-17.60 -70.00
-18.50 -71.50
IF CHINA
9
48.0 135.0
53.0 122.5
52.0 100.0
47.5 82.5
39.0 74.0
31.5 79.0
21.5 102.0
18.5 110.0
30.0 122.5
ENDIF ! This directive optional. It should never be executed.
```

Figure 4.3: Sample service area file for the ELLARCEC program.

Chapter 5

Procedures for Ordering Satellites

5.1 Motivation

It can be said that satellite synthesis problems are actually comprised of two problems:

- the problem of ordering the satellites and
- the problem of locating the satellites given some satellite ordering [5].

It is reasonable to say that a good solution strategy for satellite synthesis problems requires a good strategy for solving each of the parts of the problem. The second part of a synthesis problem, that is, the problem of locating an ordered set of satellites, can be solved by solving a linear program, one of the easier optimization problems to solve. Hence, a good strategy for ordering satellites would suggest a good approach for synthesis problems.

To this point, satellites have been ordered according to their desired locations before initiating the k-permutation algorithm to find approximate solutions to synthesis problems. This approach has worked well in most cases; however, for the larger problems it may impede the progress of the k-permutation algorithm toward a feasible solution. For this reason, the performance of the k-permutation algorithm may be enhanced if the initial satellite ordering is selected on the basis of some combination of anticipated interferences and the satellites' service arcs and desired locations, instead of on the basis of desired locations only. In the next two subsections, two methods that may be useful for ordering satellites are outlined.

5.1.1 Consecutive Spacing Method

A computer program is under development as part of this project in which a satellite positioning is computed based on a set of rules related to the location of the administrations which they serve. This satellite positioning represents an attempt by the program to satisfy spacing rules for the satellites "consecutively." The result is a set of satellite positions which can be considered an initial ordering to be input to the k-permutation algorithm. Since the k-permutation algorithm cannot exhaustively search all possible orderings, the better this initial ordering, the more likely the k-permutation algorithm will obtain a good final solution.

This consecutive spacing approach is summarized by a flow chart shown in Figure 5.1. The operations are summarized below.

INPUT Read in the data on the administrations and the permitted arcs for each.

ORDER The administrations are ordered initially by the average longitude of their test points (east to west).

PERMITTED ARCS The limits of the feasible arcs (the permitted arcs) are determined by the constraints placed on the minimum permitted elevation angle of the satellite as seen from the worst-case test point. This elevation angle is an input parameter and can be set quite conservatively in order to deliberately "over-constrain" the problem.

EASTMOST The eastmost satellite is placed at the eastmost position in the permitted arc.

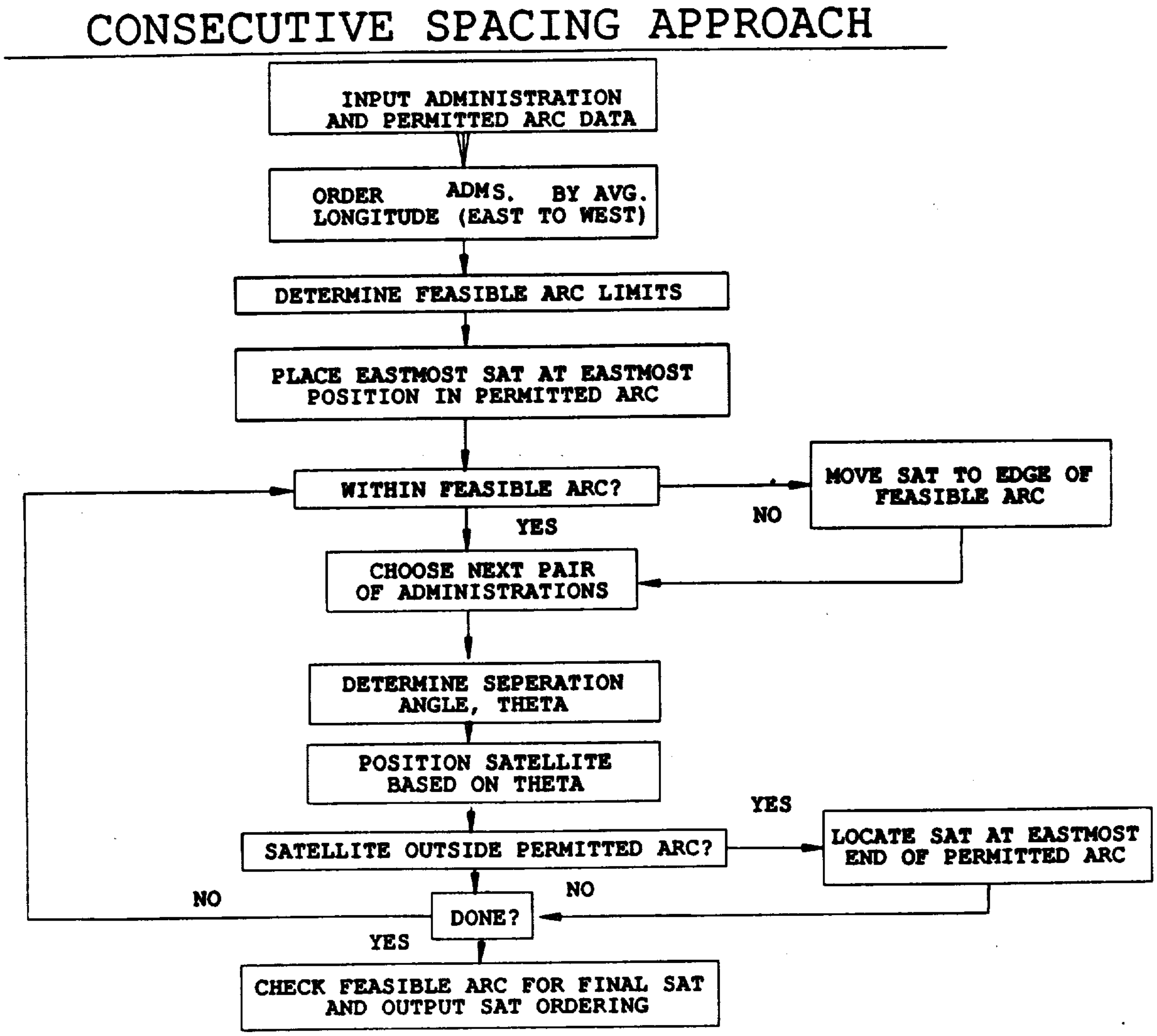

Figure 5.1: Consecutive spacing algorithm flowchart.

SHIFT Move the satellite position to within the feasible arc if necessary.

NEXT The data for the next administration is obtained.

DELTA CONSTRAINTS The positional constraints based on the DELTA-S values for previously positioned satellites are used to position each new satellite as close as possible to the eastern edge of the permitted arc. If this is not possible, the satellite is located at the eastmost end of the permitted arc.

The result of this type of positioning algorithm is that the satellites are ordered from east to west in the same order as the longitude of their administrations. When the satellites are pushed to the west-most end of their permitted arcs, they switch to the east-most end of the arcs where they might conflict with other satellites previously placed. Fortunately, the administrations are displaced in longitude by that time and closely spaced satellites should pose no problem.

An example showing the satellite positions resulting from one of the satellite positioning computer runs using this program is shown in Figure 5.2. In this case, only 22 satellites were positioned. Note the east to west progression of the positioning of the satellites until the western end of the permitted arc is reached at which time the positioning resets to the eastern edge of the permitted arcs. Remember that once the positioning is available, the results can be passed to the k-permutation algorithm as simply an initial ordering.

More work is being done on this algorithm, and results for the entire earth should be available in the near future. Comparisons between the standard k-permutation and the pre-ordered k-permutation algorithm runs will be studied – particularly in terms of the computer time required to reach a feasible solution and in the behavior of the solutions.

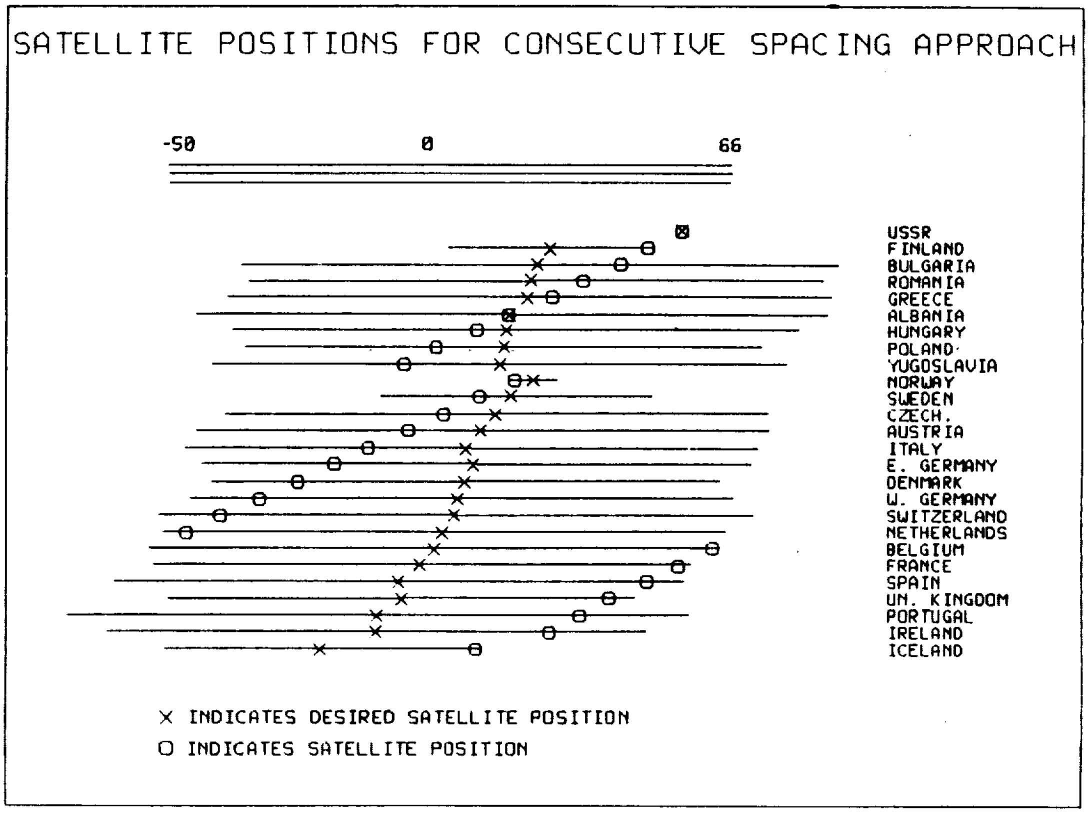

Figure 5.2: Satellite positioning resulting from consecutive spacing algorithm.

5.1.2 Slot Assignment Method

When the satellites are ordered by desired locations, there is no assurance that the satellites will be located near their desired locations nor that they will even be located within their service arcs. One goal of the method which will be referred to as the *slot assignment* method is to order the satellites in such a way that the ordering would permit each satellite to be located at a point inside its service arc.

As the slot assignment method is envisioned, the available portion of the geostationary orbit (GSO) would be divided into at least n segments of equal length, where n is the number of satellites. Then, the satellite with the shortest service arc would be allotted the segment of the GSO nearest to its desired location. Anticipated interference information would be used to indicate which segments of the arc are no longer acceptable for the remaining satellites. This process would be repeated for the satellite with the shortest remaining service arc, and so on, until all of the satellites are ordered.

The development of such a procedure will be a high-priority item in the upcoming months.

Chapter 6

Solutions to Synthesis Problems

A special-purpose heuristic procedure, a k-permutation (or switching) algorithm, has been developed under this grant for the purpose of locating geostationary satellites. The algorithm was outlined in the last interim report [6]. Though the code has been modified since that report, that algorithm remains unchanged.

The k-permutation algorithm has been exercised on six synthesis problems ranging in size from 10 to 81 satellites. The table below summarizes the results achieved with the k-permutation algorithm so far.

This table shows that the algorithm is working well on small to moderate size problems. Its lack of success on the largest problem attempted only underscores the importance of developing methods to order the satellites initially.

Interferences were calculated for the final solutions in runs where the satellite separations were calculated on the basis of total-link interference. The single-entry C/I requirement was set at 30 dB in an attempt to achieve aggregate C/I ratios of at least 25 dB. The vast majority of the aggregate C/I ratios exceeded 25 dB. There was one C/I ratio below 25 dB in the Europe(a) solution, three below 25 dB in the Europe/North Africa solution, and nine below 25 dB in the Western Hemisphere solution. Still, every aggregate C/I ratio exceeded 23 dB in spite of the fact that the objective function tends to force many of the single-entry C/I ratios to be very

Table 6.1: k-Permutation Algorithm Results

		First Feasible Solution		Final Solution	
Problem	No. of Satellites	Solution Value	CPU sec	Solution Value	CPU sec
SE Asia	10^{d}	38.18	0.4	20.18	2.3
	10^{t}	49.35	0.4	27.70	2.2
W. Europe	12^{d}	49.60	0.5	24.99	8.6
	12^{t}	68.04	0.5	31.89	3.4
Europe	$26^{a,t}$	500.96	0.9	132.45	66.6
	$26^{a,t}$	599.75	0.8	149.73	77.3
	$26^{a,t}$	525.63	0.9	141.73	145.3
Europe/N. Africa	36^{d}	601.27	1.4	135.05	555.0
	36^{t}	669.73	3.0	196.11	216.6
W. Hemisphere	50^{t}	916.73	6.3	274.58	836.1
OASTS2G1+22	81^{d}	—	—	—	5481.5

close to 30 dB.

Minor changes to the k-permutation code are planned over the next few months. The planned changes to the code will speed up execution and will reduce the variability in solution times for problems of similar size. When a final version of the program is in place, these problems will be resolved and their solutions evaluated on the basis of interference. Hopefully, additional large problems will also be solved.

Chapter 7

Arc Allotment Problems

The problem of making an allotment of an arc segment to an administration's satellite [7] can be solved using the k-permutation algorithm with modified data rather than requiring a new (or modified) algorithm. The procedure would involve the search over a scalar parameter α, which is defined as the total length (in degrees) of all allotted arcs. Also postulated is the existence of a user-specified vector of weights $(w_1, w_2, ..., w_n)$ whose elements are values assigned to each satellite. These values must be normalized to sum to one and will then represent the proportion of the total allotted arc that can be allotted to the corresponding satellites. For example, suppose $w_1 = 0.015$ and a solution is found where $\alpha = 135°$, then satellite i would receive an allotted arc segment of just over 2° , which is equal to 0.015x135°.

A solution to the problem consists of finding the largest value for α for which the corresponding modified point allotment problem has a feasible solution found by the k-permutation algorithm. The centers of the allotted arcs will then be located at the point solutions and will therefore collectively be as near to the desired locations as possible, given the limitations of the algorithm.

To accomplish this the modified point problem that needs to be solved for each candidate value of α is defined as follows:

$$\Delta'_{ij} = \begin{cases} \Delta_{ij} + \alpha(w_i + w_j)/2, & \text{if } \Delta_{ij} > 0; \\ 0, & \text{otherwise.} \end{cases}$$

$$E'_i = E_i + \alpha w_i/2$$

$$W'_i = W_i - \alpha w_i/2$$

where Δ_{ij} is the minimum required separation between satellites serving administrations i and j needed to insure meeting single-entry interference standards, and E_i and W_i are the easternmost and westernmost limits of the service arc for administration i. The k-permutation algorithm would then be used to solve a point allotment problem using the primed values in place of the unprimed values as the input data for the problem.

An ongoing investigation is aimed at developing strategies for seeking the largest candidate value of α.

Chapter 8

New Directions

In the months ahead, our primary goal is to exercise the software developed under this grant on (large) example problems. Some modifications to our existing programs are planned; all of these changes should make our package of programs easier to use and faster to execute.

Much emphasis will be placed on developing procedures to order satellites and assessing their impact on the execution time of the k-permutation algorithm and the quality of the solutions found with this procedure.

References

1. P.A. Kohnhorst, C.A. Levis and E.K. Walton, *Engineering Calculations for the "Delta S" Method of Solving the Orbital Allotment Problem*, The Ohio State University ElectroScience Laboratory, October 1987.
2. C. Wang, C. Levis and O. Buyukdura, "Optimization of Orbital Assignment and Specification of Service Areas in Satellite Communications," The Ohio State University ElectroScience Laboratory, Technical Report 716548-7 for Grant NAG 3-159, February 1987.
3. P. Bhasin and C. Reilly, "Mathematical Programming Formulations for Satellite Synthesis," The Ohio State University ElectroScience Laboratory, Technical Report 718688-4 for Grant NAG 3-159, July 1987.
4. Y. Ito, T. Mizuno and T. Muratani, "Effective Utilization of the Geostationary Orbit Through Optimization," *IEEE Transactions on Communications*, Vol. COM-27, No. 10, pp.1551-58, October 1979.
5. C. Mount-Campbell, C. Reilly and D. Gonsalvez, "A Mixed, 0-1 Integer Programming Formulation of the FSS Synthesis Problem Using Minimum Required Pair-Wise Separations," The Ohio State University, Department of Industrial and Systems Engineering, Working Paper 1986-006.
6. C. Reilly, E. Walton and P. Kohnhorst, "Engineering Calculations for Communications Satellite Systems Planning," The Ohio State University, ElectroScience Laboratory, Interim Report 718688-3 for Grant NAG 3-159, May 1987.
7. C. Reilly, "A Satellite System Synthesis Model for Orbital Arc Allotment Optimization," *IEEE Transactions on Communications*, To appear. (See also Technical Report 718688-5 for Grant NAG 3-159.)
8. D. Gonsalvez, C. Reilly and C. Mount-Campbell, "On Orbital Allotments for Geostationary Satellites," The Ohio State University ElectroScience Laboratory, Technical Report 718688-2 for Grant NAG 3-159, November 1986.