OSU

The Ohio State University

BROADCASTING SATELLITE SERVICE SYNTHESIS USING GRADIENT AND CYCLIC COORDINATE SEARCH PROCEDURES

by

Charles H. Reilly
Clark A. Mount-Campbell
David J. Gonsalvez
Clarence H. Martin
Curt A. Levis
Cou-Way Wang

The Ohio State University
Columbus, Ohio 43210

The Ohio State University

ElectroScience Laboratory

Department of Electrical Engineering
Columbus, Ohio 43212

Technical Report 716548-4
Contract NAG 3-159
February 1986

National Aeronautics and Space Administration
Lewis Research Center
21000 Brookpark Road
Cleveland, Ohio 44135

NOTICES

BROADCASTING SATELLITE SERVICE SYNTHESIS USING GRADIENT AND CYCLIC COORDINATE SEARCH PROCEDURES

by

Charles H. Reilly
Clark A. Mount-Campbell
David J. Gonsalvez
Clarence H. Martin
Curt A. Levis
Cou-Way Wang

The Ohio State University
Columbus, Ohio 43210

Technical Report 716548-4
Contract NAG 3-159
February 1986

National Aeronautics and Space Administration
Lewis Research Center
21000 Brookpark Road
Cleveland, Ohio 44135

50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. 761548-4	2.	3. Recipient's Accession No

4. Title and Subtitle Broadcasting Satellite Service Synthesis Using Gradient and Cyclic Coordinate Search Procedures	5. Report Date February 1986 6.
7. Author(s) Reilly, Mount-Campbell, Gonsalvez, Martin, Levis, Wang	8. Performing Organization Rept No 716548-4
9. Performing Organization Name and Address The ElectroScience Laboratory The Ohio State University 1320 Kinnear Road Columbus, Ohio 43212	10. Project/Task/Work Unit No. 11. Contract(C) or Grant(G) No (C) NAG 3-159 (G)
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)

Two search techniques are considered for solving satellite synthesis problems. Neither is likely to find a globally optimal solution. In order to determine which method performs better and what factors affect their performance, we design an experiment and solve the same problem under a variety of starting solution configuration-algorithm combinations. Since there is no randomization in the experiment, we present results of practical, rather than statistical, significance. Our implementation of a cyclic coordinate search procedure clearly finds better synthesis solutions than our implementation of a gradient search procedure does with our objective of maximizing the minimum C/I ratio computed at test points on the perimeters of the intended service areas. The length of the available orbital arc and the configuration of the starting solution are shown to affect the quality of the solutions found.

17. Document Analysis a. Descriptors

Satellite	Frequency
Broadcasting Service	Assignment
Fixed Service	Interference
Orbit	Experimental Design

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No of Pages
APPROVED FOR PUBLIC RELEASE: DISTRIBUTION IS UNLIMITED.	Unclassified	36
	20. Security Class (This Page)	22. Price

(See ANSI-Z39.18) See Instructions on Reverse OPTIONAL FORM 272 (4-77) (Formerly NTIS-35) Department of Commerce

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. Introduction

The Broadcasting Satellite Service (BSS) synthesis problem can be described as follows. Each broadcasting satellite must be assigned a location in the geostationary orbit and operating frequencies in the 12.2 - 12.7 GHz band for transmitting signals to its Earth receivers. An orbital arc of feasible satellite positions (feasible arc) is given for each satellite. These arcs are determined so that the following two operating requirements will be met: (1) there should be, for each satellite, a sufficient angle of elevation to permit the illumination of the entirety of the intended service area, and (2) no satellite shall be in the Earth's shadow at a time when service from that satellite is desired (eclipse protection). The objective is to assign locations and frequencies that will maintain signal interference at or below some tolerable threshold at every test point of every service area. We use the aggregate carrier-to interference (C/I) ratio at intermediate frequency (IF) for the down-link to measure the relative strength of the interfering signals. We consider acceptable any solution (specification of locations and frequencies) in which the aggregate IF C/I ratio at all test points is 30 dB or greater; this is equivalent to requiring a protection ratio of 30 dB.

A variety of approaches might be taken to obtain solutions to this synthesis problem -- from heuristic rules-of-thumb to more sophisticated mathematical programming algorithms. We have adopted the following nonlinear programming formulation of this problem [4,6]:

$$\text{Minimize } Z = \sum_{i} \sum_{j \varepsilon J_i} \sum_{k \varepsilon K_i} \exp\,(\alpha - (C/I)_{ijk}) \tag{1}$$

$$\text{subject to} \quad e_i \leqslant x_i \leqslant w_i \ , \quad \forall i \tag{2}$$

$$l_i \leqslant y_i \leqslant h_i \ , \quad \forall i \tag{3}$$

where x_i is the location of satellite i, y_i is the first (lowest) frequency in the family of channels assigned to satellite i, e_i (w_i) is the easternmost (westernmost) feasible location for satellite i, l_i(h_i) is the lowest (highest) frequency that can be assigned to satellite i, α is a scaling constant, J_i is the set of test points in the service area served by satellite i, and K_i is the set of frequencies at which signals are transmitted from satellite i.

The computation of the C/I ratios and the objective function value Z is much more complicated than it might appear to be when one examines this formulation. C/I is actually a function of topocentric and satellite-centered angles, frequency discrimination, antenna gains and discrimination patterns, elliptical beam parameters, and transmitted power. For brevity, the complete expression for C/I is not presented here. We treat C/I as a function of locations and frequencies only [4,6].

This formulation seems quite simple. The constraints (2) and (3) serve only to bound the values of the decision variables x_i and y_i. The objective function (1) is intended to maximize the smallest C/I ratio at any test point. The use of the exponential function should result in the accentuation of the most unfavorable C/I ratio.

This synthesis problem, despite its simple formulation, is a difficult optimization problem. The objective function surface is not known to possess any properties which might be easily exploited by a special-purpose algorithm. This surface has many peaks, ridges, and valleys of different heights and depths. There will be no practical way to verify that a solution is a global optimum; most likely, any solution found will be a local optimum.

The formulation of the BSS synthesis problem and the techniques we suggest for solving it are quite likely to be applicable to the Fixed Satellite Service (FSS) synthesis problem as well. The calculation of the C/I ratios would have to be modified to account for the up-link as well as the down-link. At this time, the synthesis of FSS satellites is of greater interest. However, no cross-polarized antenna discrimination patterns for FSS antennas have been recommended as yet. We felt that our results would be more easily extrapolated to the complete FSS synthesis problem if we used co-polarized and cross-polarized BSS antenna patterns in our test problem than if we used the co-polarized FSS antenna patterns alone.

In the next section we describe two search techniques which we have used to solve the synthesis problem as formulated here. In the third section, a small test problem and a carefully designed plan for evaluating these search techniques with the test problem are presented. The results of the experiment are presented in section IV. Finally, our conclusions are discussed in section V.

II. Solution Methods Considered

In this section, we present two search techniques which can be used to solve the BSS synthesis problem. For unconstrained optimization problems with a unique optimum, both of these methods are certain to converge to the global optimum [5]. The same is not true for the problem we are dealing with here. However, there is no reason to expect that either of these search methods can not produce good solutions (local optima) to our synthesis problem.

Gradient Search Technique

The gradient search technique, or the method of steepest descent, is a procedure designed to find globally optimal solutions to unconstrained convex programming problems. The problem we have here is not a convex programming problem; hence, we can not expect to find a global minimum. Furthermore, our problem includes constraints (2) and (3) which enforce upper and lower bounds on the location and frequencies assigned to the satellites being considered. Collectively, these constraints form a hypercube. The gradient search procedure can be modified in a variety of ways in order to recognize the presence of these restrictions on the decision variables.

We have chosen to modify the procedure in a way that maintains simplicity in the algorithm, but that can cause it to converge to boundary points that are not even locally optimal. This particular trade-off in the design of our implementation of the algorithm was arrived at for a number of reasons, and it is the determination of the cost of this trade-off (in terms of the quality of solutions) that is one motivation for the experiment we describe later.

A significant portion of the calculations in a gradient search technique are devoted to computing the gradient of the objective function, ∇Z, with respect to the decision variables in order to find vector directions in which to conduct line searches for better solutions. The calculation of the gradient in the case of this BSS synthesis problem is rather involved. Parts of the calculation are carried out analytically, but others are done numerically. Once the gradient is computed, the line search is conducted in the negative gradient direction until a solution more attractive than the current one is found or until a fairly extensive line search yields no improved solution. Only solutions that are within the hypercube of feasible solutions are examined.

We can formally state our implementation of this algorithm as:

Step 1: Choose a feasible starting solution,
(x_0, y_0). Let $m = 0$.

Step 2: If $m > m^*$, go to Step 5. Otherwise, compute $\nabla Z(x_0, y_0)$.
If $\nabla Z(x_0, y_0) = 0$, go to Step 5.
Otherwise
let $n = 0$ and $m = m + 1$.

Step 3: Let $n = n + 1$. Determine the length, d, of the line segment from (x_0, y_0) to the nearest boundary in the $-\nabla Z(x_0, y_0)$ direction. Evaluate the objective function (1) at ten equally spaced feasible points in the $-\nabla Z$ direction,

$$(x,y) = (x_0,y_0) - \frac{(.1)kd}{(5)^n} \nabla Z(x_0,y_0), \qquad k = 1, 2,.., 10.$$

Select the most favorable of these ten points, (x^*,y^*). If $Z(x^*,y^*) < Z(x_0,y_0)$, set $(x_0,y_0) = (x^*,y^*)$ and go to Step 2. Otherwise, go to Step 4.

Step 4: If $n > 12$, go to Step 5. Otherwise go to Step 3.

Step 5: Stop.

This algorithm terminates either when the gradient vanishes, when 13 searches of 10 points each in the $-\nabla Z$ direction fail to produce an improved solution, or when m^* (user specified) iterations are completed.

In any search technique, a direction in which to search has to be selected. In the case of the gradient search procedure, the gradient, or direction of greatest instantaneous change, evaluated at a point, is chosen. It may be that this is not a good choice in our case. Our objective function surface has many peaks and valleys and there is no reason to presume that the local minimum toward which $-\nabla Z$ points is an attractive one. Furthermore, during the line search conducted in the

$-\nabla Z$ direction, we examine candidate solutions often at points quite far removed from that at which the gradient was evaluated. Therefore, we elected to consider a second search procedure that avoids using the gradient to help determine a search direction.

Cyclic Coordinate Search Technique

Like the gradient search technique, the cyclic coordinate search algorithm is globally convergent when solving unconstrained convex programming problems. It can also be modified in a number of ways to account for the upper and lower bound constraints present in our problem. Again, we chose a simple modification that does not, in general, guarantee convergence to a local optimum.

This algorithm begins with the selection of a feasible point. Line searches are conducted, in turn, in each of the positive and negative coordinate directions. The only computationally intensive calculations are the many evaluations of the objective function performed during these line searches.

Formally, we can state the algorithm as follows:

Step 1: Choose a feasible starting solution, (x_0, y_0). Let $m = 0$.

Step 2: If $m \geq m^*$, go to Step 6. Otherwise, let $n = 1$ and $m = m+1$.

Step 3: Select a coordinate direction corresponding to a location variable x_i. Evaluate the objective function (1) at 5 feasible points in both the positive and negative coordinate directions:

$$(x,y_0) = (x_0,y_0) + (.2)k\left(0,\ldots,0, \frac{w_i - x_{io}}{3^{n-1}}, 0,\ldots,0\right);\ k = 1,2,3,4,5$$

$$(x,y_0) = (x_0,y_0) - (.2)k\left(0,\ldots,0, \frac{x_{io} - e_i}{3^{n-1}}, 0,\ldots,0\right);\ k = 1,2,3,4,5$$

Choose the best of these points, (x^*,y^*).

If $Z(x^*,y^*)<Z(x_0,y_0)$, set $(x_0,y_0) = (x^*,y^*)$. Repeat this step for another coordinate direction corresponding to a different location variable until all of these directions have been considered.

Step 4: Select a coordinate direction corresponding to a frequency variable y_i. Evaluate the objective function (1) at 5 feasible points in both the positive and negative coordinate directions:

$$(x_0,y) = (x_0,y_0) + (.2)k\left(0,\ldots,0, \frac{h_i - y_{io}}{3^{n-1}}, 0,\ldots,0\right);\ k = 1,2,3,4,5$$

$$(x_0,y) = (x_0,y_0) - (.2)k\left(0,\ldots,0, \frac{y_{io} - l_i}{3^{n-1}}, 0,\ldots,0\right);\ k = 1,2,3,4,5$$

Choose the best of these points (x^*,y^*).

If $Z(x^*,y^*)<Z(x_0,y_0)$, set $(x_0,y_0) = (x^*,y^*)$. Repeat this step for another coordinate direction corresponding to a different frequency variable until all of these directions have been considered. If no improved solutions are found in Steps 3 and 4 and $n \leq 4$, set $n=n+1$ and repeat these steps. Otherwise, go to Step 5.

Step 5: If no improved solution was found in the current cycle (iteration), go to Step 6. Otherwise, go to Step 2.

Step 6: Stop.

This search algorithm terminates when 5 searches of 5 points in each of the positive and negative coordinate directions fail to yield an improved solution or when m* (user specified) cycles (iterations) are completed.

With the cyclic coordinate search, there are likely to be many more functional evaluations of the objective function than there would be with the gradient search. However, the gradient of the objective function, which represents a substantial computational burden, is never computed. During every line search with the cyclic coordinate search, points are examined from one feasible extreme of a coordinate direction to the other feasible extreme. The line searched in the gradient search procedure can be quite short if the current solution is near a boundary of the feasible region. Therefore, there is always the possibility that the gradient search algorithm will become trapped near a boundary. Because the cyclic coordinate search algorithm examines candidate solutions from boundary to boundary in each direction, it has more opportunity to move away from a nearby boundary.

III. Plan For Evaluating The Search Techniques

We designed an experiment to enable us to assess the performance of the gradient and cyclic coordinate search algorithms as synthesis tools. These algorithms were exercised under a variety of conditions, e.g., different starting solutions, different feasible regions, and different run lengths. In this way we are able not only to report which technique seems to perform better but which factors tend to affect performance most.

The synthesis test problem we considered consists of seven South American countries (Argentina, Bolivia, Brazil, Chile, Paraguay, Peru, and Uruguay), each to be served by one broadcasting satellite. Each of the satellites is assumed to be capable of transmitting signals over a family of three adjacent channels with alternating polarization. Decision variables need only be defined for the lowest frequency assigned to each satellite. The remaining two frequencies are then the next two higher frequencies. All seven satellites are assigned the same feasible orbital arc and feasible spectrum.

A list of the factors or variables that we believe are most likely to affect the performance of these algorithms was compiled. A total of eight factors, including a factor for algorithm, were listed. Two levels, one low and one high, were specified for each factor. The worst aggregate C/I ratio at any test point was selected as the criterion to evaluate the results. The factors and factor levels included in our experiment are shown in Table 1.

Each of the factors was included for a particular reason. ALGORITHM was included as a factor so that we can evaluate both of the search procedures as synthesis tools. A solution in which the satellites are collocated and the satellites utilize the same frequencies is certain to be a poor solution. By including the LOCATION SPACING and FREQUENCY SPACING factors, we will have some starting solutions that would be most unattractive as final solutions as well as some more attractive ones. Boundaries of the feasible region are known to affect the performance of the gradient search algorithm. The STARTING POSITIONS and STARTING FREQUENCIES factors allow us to specify initial solutions located on a boundary of the feasible region. Since the geostationary orbit and the available spectrum are limited resources, conserving the orbit and spectrum is useful. The factors ARC LENGTH and FREQUENCY SPECTRUM allow us to experiment with a shorter arc and/or a narrower spectrum. RUN TIME is included because it is doubtful that true local minima will be found. By extending the length of some runs, it may be possible to obtain significantly better solutions.

We are not only interested in estimating the individual main effects of these factors, but in the factor interactions as well. By making 256 computer runs, we could observe the solutions obtained for all possible combinations of factor levels and estimate all of the main effects and factor interactions. We elected instead to make 64 runs, using a 1/4 - fractional factorial design [1,2]. In this way, we are

TABLE 1

FACTORS AND FACTOR LEVELS

Factor	Factor Levels: Low (-1)	Factor Levels: High (+1)
A - Algorithm	gradient search	cyclic coordinate search
B - Location Spacing	0°	1°
C - Frequency Spacing	0 MHz	5 MHz
D - Starting Locations	centered in feasible arc	spaced from westernmost feasible location
E - Starting Frequencies	centered in available spectrum	spaced from highest available frequency
F - Arc Length	90 - 110°W	80 - 110°W
G - Frequency Spectrum	12233-12300 MHz	12200-12300 MHz
H - Run Time	5 CPU minutes or 10 iterations or cycles	10 CPU minutes or 30 iterations or cycles

Source: Authors' Assumptions.

still able to estimate the main effects and all of the two-factor interactions. Other interaction terms can also be estimated, but in most instances we are assuming that interactions of third or higher order are negligible.

When a fractional factorial design is used in an experiment, there is usually some random element that allows an experimenter to summarize the results of the experiment in the form of probability statements and to quantify the statistical significance of the results. There is no random element included in our experiment, yet we will employ the same method of analysis that would be used if a random element were present. It is not our intention to make probability statements; our concern is with practical, rather than statistical, significance.

IV. Results Of The Experiment

The 64 computer runs are summarized in Table 2. Columns A through H indicate the levels of the eight factors used in each run. An entry of -1 indicates the factor was used at the low level, while an entry of +1 indicates the factor was used at the high level. The last four columns show the worst aggregate C/I ratio in the final solution found, the number of iterations or cycles completed, the expired computing time (in CPU seconds) at the end of the last completed iteration or cycle, and the reason the run terminated, respectively, for each run. If the iteration/cycle limit was reached, an entry of one appears in the last column. An entry of two indicates the allotted time had expired. Finally, if a three is entered in this column, then all attempts made to

TABLE 2

COMPUTER RESULTS

Run	A	B	C	D	E	F	G	H	Worst C/I	Iterations or Cycles	Expired CPU Time	Stop Type
1	-1	-1	-1	-1	-1	-1	-1	-1	13.70	10	64	1
2	+1	+1	-1	-1	-1	-1	-1	+1	35.95	13	547	2
3	+1	+1	-1	-1	-1	-1	+1	-1	37.73	10	294	1
4	-1	-1	-1	-1	-1	-1	+1	+1	13.88	12	77	3
5	+1	-1	-1	-1	-1	+1	-1	-1	41.47	7	275	2
6	-1	+1	-1	-1	-1	+1	-1	+1	24.66	3	13	3
7	-1	+1	-1	-1	-1	+1	+1	-1	24.66	3	12	3
8	+1	-1	-1	-1	-1	+1	+1	+1	49.13	13	558	2
9	-1	+1	-1	-1	+1	-1	-1	-1	17.34	10	67	1
10	+1	-1	-1	-1	+1	-1	-1	+1	36.78	13	538	2
11	+1	-1	-1	-1	+1	-1	+1	-1	39.99	10	280	1
12	-1	+1	-1	-1	+1	-1	+1	+1	21.62	30	196	1
13	+1	+1	-1	-1	+1	+1	-1	-1	44.01	9	287	2
14	-1	-1	-1	-1	+1	+1	-1	+1	30.22	27	280	3
15	-1	-1	-1	-1	+1	+1	+1	-1	24.92	10	60	1
16	+1	+1	-1	-1	+1	+1	+1	+1	52.93	13	321	3
17	-1	+1	-1	+1	-1	-1	-1	-1	5.07	3	16	3
18	+1	-1	-1	+1	-1	-1	-1	+1	36.37	16	585	2
19	+1	-1	-1	+1	-1	-1	+1	-1	35.75	10	276	1
20	-1	+1	-1	+1	-1	-1	+1	+1	20.33	12	76	3
21	+1	+1	-1	+1	-1	+1	-1	-1	43.63	9	256	2
22	-1	-1	-1	+1	-1	+1	-1	+1	28.43	30	117	1
23	-1	-1	-1	+1	-1	+1	+1	-1	28.43	10	37	1
24	+1	+1	-1	+1	-1	+1	+1	+1	51.79	17	554	2
25	-1	-1	-1	+1	+1	-1	-1	-1	24.41	6	24	3
26	+1	+1	-1	+1	+1	-1	-1	+1	35.19	13	539	2
27	+1	+1	-1	+1	+1	-1	+1	-1	41.20	9	297	2
28	-1	-1	-1	+1	+1	-1	+1	+1	24.40	30	209	1
29	+1	-1	-1	+1	+1	+1	-1	-1	45.71	10	300	1
30	-1	+1	-1	+1	+1	+1	-1	+1	6.59	30	322	1
31	-1	+1	-1	+1	+1	+1	+1	-1	10.15	3	12	3
32	+1	-1	-1	+1	+1	+1	+1	+1	49.99	17	572	2

TABLE 2 (continued)

Run	A	B	C	D	E	F	G	H	Worst C/I	Iterations or Cycles	Expired CPU Time	Stop Type
33	+1	-1	+1	-1	-1	-1	-1	-1	34.76	6	253	2
34	-1	+1	+1	-1	-1	-1	-1	+1	10.44	9	56	3
35	-1	+1	+1	-1	-1	-1	+1	-1	18.83	10	62	1
36	+1	-1	+1	-1	-1	-1	+1	+1	39.31	17	552	2
37	-1	-1	+1	-1	-1	+1	-1	-1	21.53	9	63	3
38	+1	+1	+1	-1	-1	+1	-1	+1	44.06	19	585	2
39	+1	+1	+1	-1	-1	+1	+1	-1	43.32	10	278	1
40	-1	-1	+1	-1	-1	+1	+1	+1	35.81	11	31	3
41	+1	+1	+1	-1	+1	-1	-1	-1	37.94	8	254	2
42	-1	-1	+1	-1	+1	-1	-1	+1	14.57	30	160	1
43	-1	-1	+1	-1	+1	-1	+1	-1	14.57	10	57	1
44	+1	+1	+1	-1	+1	-1	+1	+1	40.27	16	551	2
45	-1	+1	+1	-1	+1	+1	-1	-1	28.98	10	39	1
46	+1	-1	+1	-1	+1	+1	-1	+1	47.16	14	359	3
47	+1	-1	+1	-1	+1	+1	+1	-1	53.58	10	225	1
48	-1	+1	+1	-1	+1	+1	+1	+1	28.98	30	118	1
49	+1	+1	+1	+1	-1	-1	-1	-1	31.73	7	288	2
50	-1	-1	+1	+1	-1	-1	-1	+1	24.07	30	207	1
51	-1	-1	+1	+1	-1	-1	+1	-1	35.68	10	64	1
52	+1	+1	+1	+1	-1	-1	+1	+1	34.89	15	583	2
53	-1	+1	+1	+1	-1	+1	-1	-1	2.16	3	16	3
54	+1	-1	+1	+1	-1	+1	-1	+1	37.75	14	573	2
55	+1	-1	+1	+1	-1	+1	+1	-1	46.65	10	198	1
56	-1	+1	+1	+1	-1	+1	+1	+1	16.79	8	43	3
57	+1	-1	+1	+1	+1	-1	-1	-1	33.76	9	298	2
58	-1	+1	+1	+1	+1	-1	-1	+1	2.88	9	122	3
59	-1	+1	+1	+1	+1	-1	+1	-1	2.88	9	103	3
60	+1	-1	+1	+1	+1	-1	+1	+1	38.01	16	588	2
61	-1	-1	+1	+1	+1	+1	-1	-1	27.13	10	111	1
62	+1	+1	+1	+1	+1	+1	-1	+1	45.15	12	302	3
63	+1	+1	+1	+1	+1	+1	+1	-1	46.38	7	181	3
64	-1	-1	+1	+1	+1	+1	+1	+1	27.13	11	124	3

Source: Authors' Experimental Design and Computer Program Calculations.

find a solution better than the final reported solution failed. All of these runs were made on an IBM 3081 computer at The Ohio State University.

The initial and final solutions for two of the runs, 47 and 53, are displayed in Table 3. These runs provided the best and worst solutions as measured by worst aggregate C/I ratio, respectively. We can see that the cyclic coordinate algorithm produced a solution in Run 47 which uses the entire feasible orbital arc (80-110°W) and the entire available frequency spectrum (12200 - 12300 MHz). The final solution obtained exceeds the acceptable C/I protection ratio by over 23 dB even though the initial solution was horrendous; all of the satellites were collocated at the start.

The initial solution for Run 53 was not nearly an acceptable solution, although it is better than the starting solution for Run 47. In this case, the gradient search algorithm was not able to find a solution significantly better than the initial solution. The locations and frequencies changed very little during this run. Presumably, the search for improved solutions was hindered because the initial solution was located on a boundary of the feasible region.

We could show similar tables for the 62 remaining runs, but have chosen not to in the interest of space. Two important points can be made from Table 3 alone, however. First of all, better final solutions do not necessarily result from better starting solutions. Secondly, the cyclic coordinate procedure is more likely to use the entirety of the feasible arc and available spectrum because the line search conducted in

TABLE 3

EXAMPLE SOLUTIONS

Country		Run 47 Initial	Run 47 Final	Run 53 Initial	Run 53 Final
Argentina	Loc.	95°W	80°W	104°W	100.89°W
	Freq.	12270 MHz	12200 MHz	12235 MHz	12233.38 MHz
Bolivia	Loc.	95°W	110°W	105°W	102.98°W
	Freq.	12275 MHz	12300 MHz	12240 MHz	12237.93 MHz
Brazil	Loc.	95°W	110°W	106°W	109.35°W
	Freq.	12280 MHz	12240.74 MHz	12245 MHz	12245.75 MHz
Chile	Loc.	95°W	88.28°W	107°W	107.91°W
	Freq.	12285 MHz	12252.44 MHz	12250 MHz	12248.56 MHz
Paraguay	Loc.	95°W	96.94°W	108°W	107.97°W
	Freq.	12290 MHz	12300 MHz	12255 MHz	12261.18 MHz
Peru	Loc.	95°W	96.96°W	109°W	109.53°W
	Freq.	12295 MHz	12200 MHz	12260 MHz	12261.41 MHz
Uruguay	Loc.	95°W	80°W	110°W	110.00°W
	Freq.	12300 MHz	12300 MHz	12265 MHz	12261.50 MHz
	Worst C/I	-4.82 dB	53.58 dB	0.67 dB	2.16 dB

Sources: Authors' Solutions.

each coordinate direction extends from one feasible extreme to the other. The rest of the results are summarized below.

Summary of Results

Recall that our primary objective was to assess the performance of the gradient and cyclic coordinate search procedures as tools for BSS synthesis. The criterion we have chosen to evaluate these techniques is the smallest C/I ratio at any test point. We also would like to determine what other factors, besides ALGORITHM, affect the performance of these techniques. Solution times and rates of improvement also deserve consideration.

By ranking the 64 runs made in descending order of worst C/I (See Table 2), we find that 32 of the 34 best results are obtained by using the cyclic coordinate procedure. The worst aggregate C/I values for the cyclic coordinate runs range from 31.73 to 53.58 dB, while the corresponding range for the gradient search runs extends from 2.16 to 35.81 dB. For the ranked results, results numbered 27, 29, and 35 through 64 are produced by gradient search runs. All of the cyclic coordinate solutions satisfy the criterion for acceptibility as the smallest C/I ratio exceeds the threshold of 30 dB in every case. Only three of the gradient search solutions can be considered acceptable by this standard.

Over all of the runs made, the average worst C/I was 30.68 dB. The average for gradient search runs was 19.73 dB, 18.78 dB for the shorter runs and 20.67 dB for the longer runs. For the cyclic coordinate runs, the averages were 41.10 dB for short runs, 42.17 dB for long runs, and 41.64 dB overall. There seems to be little advantage to making long cyclic coordinate runs since short runs seem to find acceptable solutions quickly. However, only three of these longer runs terminated before the limits of 10 CPU minutes or 30 cycles. Further improvement may have been possible.

It is interesting to examine the reasons for run termination, i.e., iteration count (stop type 1), time (stop type 2), no improved solution found (stop type 3) by algorithm and run length. Stops of type 2 only occurred during cyclic coordinate runs. None of the longer (10 CPU minutes or 30 cycles) cyclic coordinate runs experienced a stop of type 1. Apparently, more than 10 CPU minutes is required to complete 30 cycles. A total of 20 runs were terminated by a stop of type 3; 16 of these 20 runs were gradient search runs. Of the four cyclic coordinate runs which experienced a type 3 termination, the poorest solution yielded a smallest C/I of 45.15 dB. Three of the gradient runs with type 3 termination produced very poor solutions, less than 3 dB for the worst C/I ratio. Certainly better solutions were available, but the gradient search homed in on poor ones.

Some statistics on the rates of progress and execution times for these two methods are displayed in Table 4. These statistics indicate that more improvement in solutions occurs in the early iterations or cycles. As time passes and it becomes more difficult to find an improved solution, the cycles or iterations take more time, particularly for the cyclic-coordinate technique. Per CPU second, the gradient

TABLE 4

RATES OF PROGRESS AND EXECUTION TIMES

Algorithm	Run Length	Improvement per Iteration or Cycle	CPU Seconds per Iteration or Cycle	Improvement Per CPU Second
Gradient Search	Short	2.70 dB	6.40	0.42 dB
	Long	1.19 dB	6.89	0.17 dB
	All	1.62 dB	6.75	0.24 dB
Cyclic Coordinate	Short	4.95 dB	30.07	0.16 dB
	Long	3.72 dB	43.27	0.09 dB
	All	4.23 dB	37.68	0.11 dB

Source: Authors' Calculations.

search method produces more improvement in solutions than the cyclic coordinate method only because a typical cycle requires about 5.6 times as much CPU time as a gradient search iteration requires. Even though the cyclic coordinate technique consumes more CPU time, the additional investment brings a better solution as a reward.

The analysis described so far clearly points out that the results obtained are strongly influenced by the ALGORITHM used to produce the results (factor A). After partitioning the runs by algorithm and ranking the solutions in each set in descending order of worst C/I ratio, it also becomes clear that ARC LENGTH (factor F) influences the results of a run. Among the ranked results, the best 15 cyclic coordinate runs and 11 of the 12 best gradient search runs were made with the longer arc length, 80-110°W.

Factor G (FREQUENCY SPECTRUM) is likely to have an effect similar to that of ARC LENGTH, but apparently this effect is not as pronounced in our test problem. We can see some evidence of this if we look at the best solution found for each combination of factor levels of factors F and G. The most favorable worst C/I ratios for these combinations are 37.94 dB (F = -1, G = -1), 41.20 dB (F = -1, G = +1), 47.16 dB (F = +1, G = -1), and 53.58 dB (F = +1, G = +1). Each of these solutions is considered acceptable, but, by using a broader frequency spectrum or a longer arc, we can improve upon the solution with 37.94 dB as the smallest C/I ratio. The improvement due to lengthening the arc seems to be greater than the improvement observed when the spectrum is broadened. The best solution is found with the longer arc and broader spectrum. We see that conservation of the arc and/or spectrum does affect the quality of the best solution obtained. Not surprisingly, all four solutions were found by the cyclic coordinate algorithm.

In our 64 runs, we have actually tried to solve each of 4 synthesis problems in 16 different ways by varying the solution technique used and the configuration of the initial solution. Each of the four synthesis problems is defined by the available arc (short or long) and available spectrum (narrow or broad). The four solutions mentioned above are significant because they provide lower bounds on the quality of the solutions that can be found for each problem.

We have been able to assess the impact of the factors A and F by inspection due to their obviously strong influence on our results. A more formal analysis of all factors and factor interactions is described below.

Analysis of Factor Effects and Interactions

Because we chose to use a fractional factorial design for our experiment, each factor effect and each interaction is confounded with three other factor effects and/or interactions, called aliases. Hence, an observed phenomenon could be attributed to any one, or a combination, of four aliased effects. We will adopt the convention of attributing influence to the effect of lowest order. Our experiment was designed so that no first and second order effects are confounded with other first and second order effects.

The defining contrast for our experiment is :

$$I = -ACFGH = -BDEGH = ABCDEF \tag{4}$$

I stands for the overall mean. In this case, we see that we can not distinguish the effect of the overall mean from those of its aliases, two fifth order interactions (ACFGH and BDEGH) and a sixth order interaction (ABCDEF). To find the aliases of an effect of interest, we can multiply our defining contrast (4) by that effect and treat all squared terms and I as 1. For example, suppose we wish to find the aliases of the main effect of factor E. After multiplication, we have

$$EI = -ACEFGH = -BDE^2GH = ABCDE^2F \tag{5}$$

or

$$E = -ACEFGH = -BDGH = ABCDF \tag{6}$$

We see that E is confounded with interaction terms of orders four, five, and six.

We used Yates' method for two-level experiments [1] to determine which factor effects and interactions had the greatest impact on our computer results. We found the eight effects shown in Table 5 to be the most significant.

It is not unusual for the overall mean to be one of the more significant effects. Factor A (ALGORITHM) provides the second most important effect. In light of the results presented earlier about the relative performance of the two synthesis techniques considered, this is not suprising. Similarly, the obvious impact of ARC LENGTH (factor F)

TABLE 5

MOST SIGNIFICANT EFFECTS - YATES' METHOD

Rank	Significant Effects and		Their Aliases*		Average Effect
1	I	= -ACFGH	= -BDEGH	= ABCDEF	30.68 dB
2	A	= -CFGH	= -ABDEGH	= BCDEF	21.91 dB
3	F	= -ACGH	= -BDEFGH	= ABCDE	7.97 dB
4	ABD	= -BCDFGH	= -AEGH	= CEF	5.35 dB
5	BD	= -ABCDFGH	= -EGH	= ACEF	-4.60 dB
6	AB	= -BCFGH	= -ADEGH	= CDEF	4.58 dB
7	B	= -ABCFGH	= -DEGH	= ACDEF	-4.58 dB
8	G	= -ACFH	= -BDEH	= ABCDEFG	4.26 dB

* - The aliases are assumed to have negligible influence.

Source: Authors' Calculations.

was pointed out previously. Factors B (LOCATION SPACING) and G (FREQUENCY SPECTRUM) are two additional important main effects.

Two of the two-factor interactions are significant also. The interaction BD (LOCATION SPACING, STARTING LOCATIONS) has a negative effect. Figure 1 contains a plot which illustrates this interaction. When the starting locations are centered in the feasible arc (D low), the initial spacing of the satellite locations (B) has little effect on the average of the observed worst C/I ratios. However, when the satellites are located near the western boundary of the feasible arc (D high), we find a better solution when the satellites are collocated (B low) than when they are separated by 1° (B high). The effect of this interaction is negative because the result is less attractive as we move from the low level of B to the high level of B. Presumably, when the satellites are not collocated at the western boundary but are separated by 1° (B high), the easternmost most satellites tend to "block" the progress of the other satellites which would prefer to move east.

Figure 2 can be used to explain the important second order interaction between factors A (ALGORITHM) and B (LOCATION SPACING). We see that when the gradient search algorithm (A low) is used, better solutions are found when the satellites are initially collocated (B low) than when they are separated by 1° (B high). the initial locations of the satellites has no effect on the quality of the solutions found when the cyclic coordinate algorithm is utilized (A high). This is an important finding because it indicates that we need not be concerned with the initial separation of the satellites when the cyclic coordinate

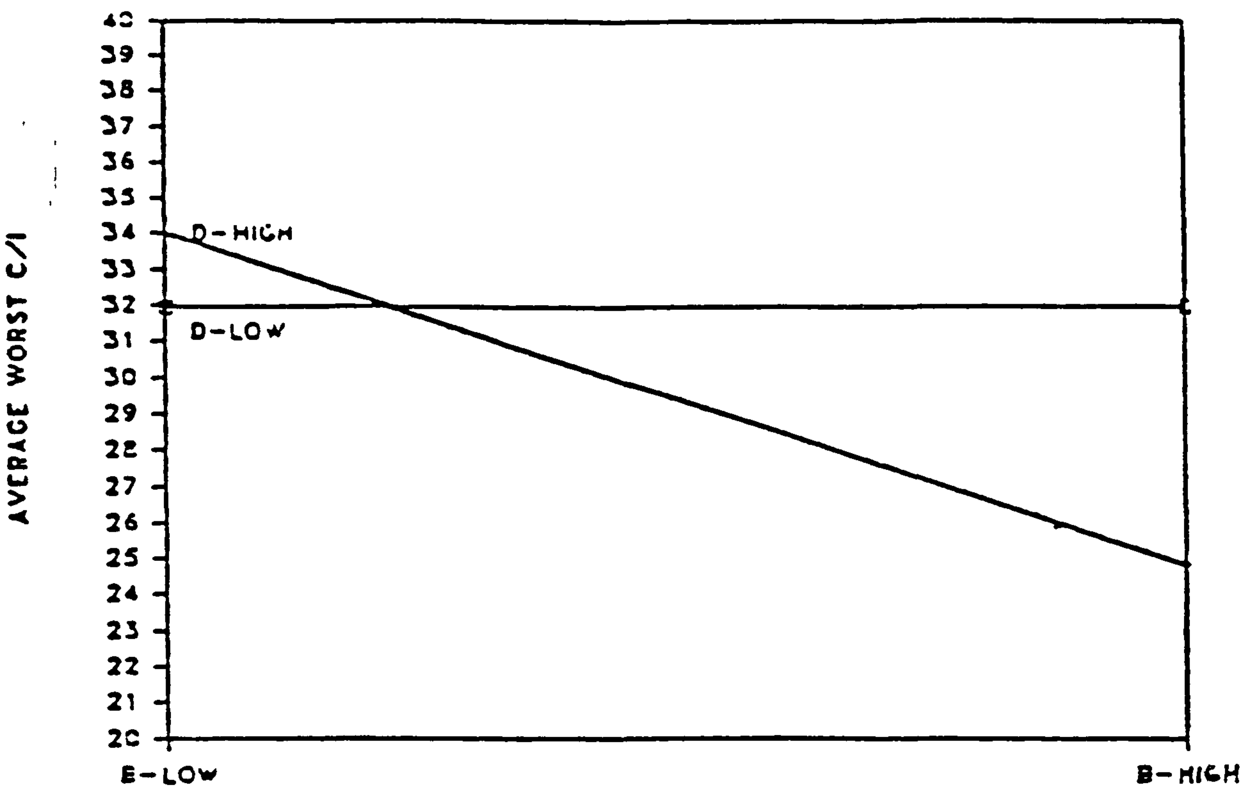

Figure 1. Plot of BD interaction.

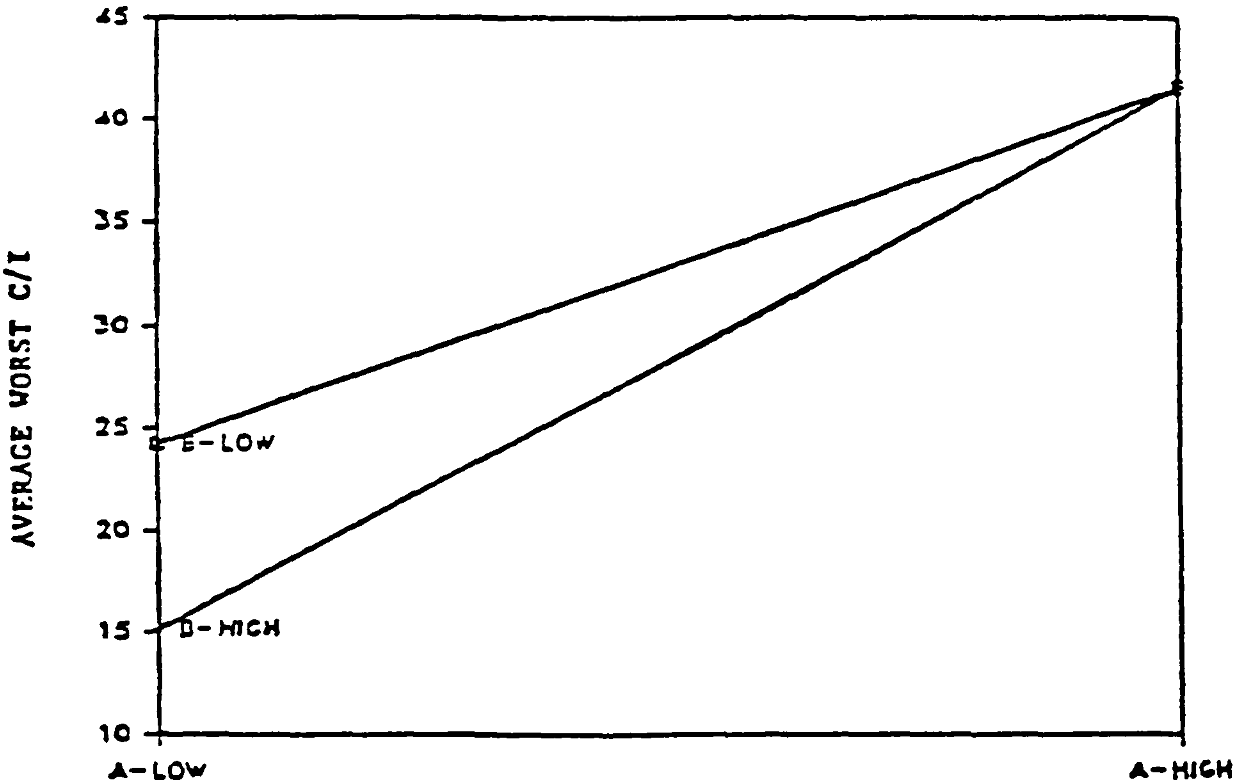

Figure 2. Plot of AB interaction.

procedure is employed. Seemingly, any set of feasible initial locations will suffice with this method.

At least one third order interaction influences our results. In Table 5, we see that the fourth most important effect is either ABD, CEF, or both. Since we have found that A, B, AB, and BD all influence our results, we will assume that the actual effect is ABD. (The main effect of factor D was the tenth most significant effect according to our ranking.) This interaction can be explained in a fairly straightforward manner. See Figure 3 for a plot of this interaction. We have already discussed the "blocking" effect in our explanation of the BD iteraction. This blocking effect is more pronounced when the gradient search procedure (A low) is used than when the cyclic coordinate procedure (A high) is used. We believe this occurs because the cyclic coordinate procedure attempts to reposition satellites one at a time and is capable of moving satellites well beyond the blocking satellites near the boundary of the feasible orbital arc. Much less movement is expected with the gradient search procedure. Some components of the gradient will indicate that moves toward the boundary look most promising. The result will be that the line segment searched in the negative gradient direction will be quite short, thereby prohibiting substantial repositioning of the satellites.

This problem with the gradient is perhaps a symptom of our having kept the algorithm as simple as possible by modifying only the line search to recognize the constraints on the possible solutions. An alternate, but more complex, approach would be to modify both the line

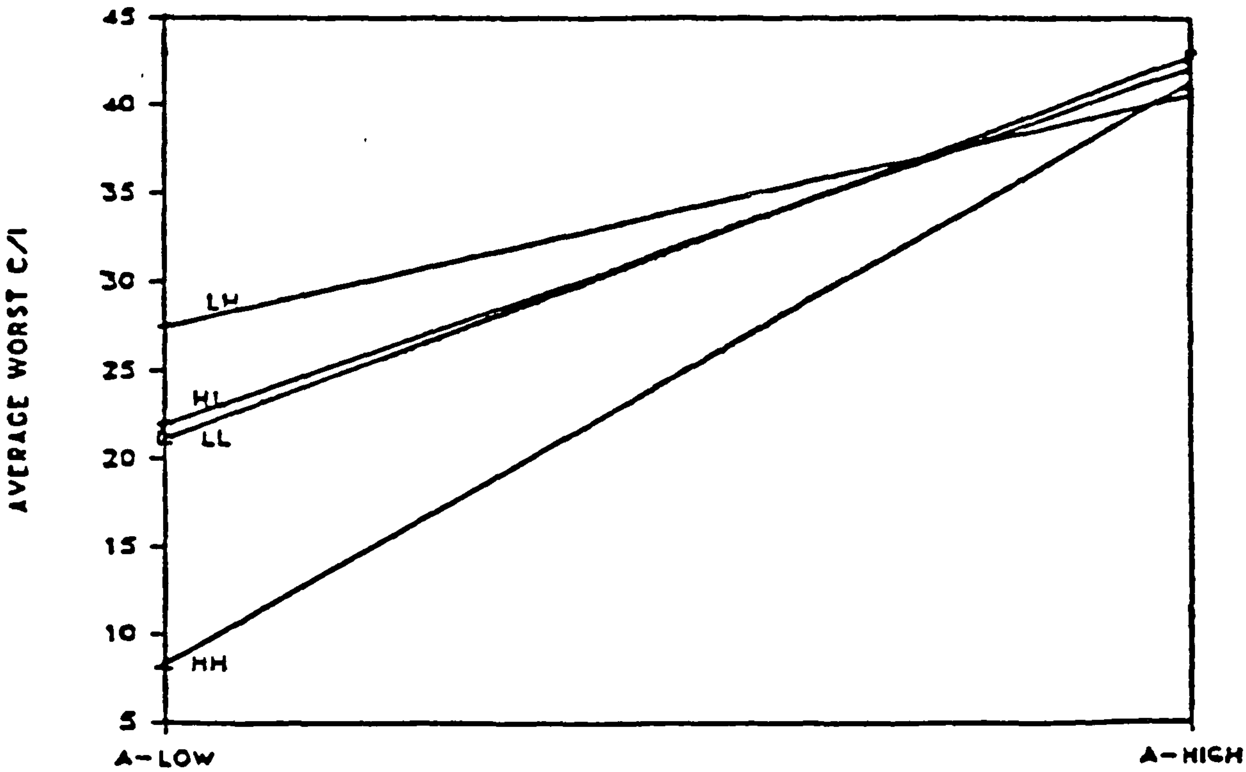

Figure 3. Plot of ABD interaction.

search and the search direction. This could prevent the gradient search from getting "hung up" on nonoptimal points on or near a boundary. However, as our experiments show, the cyclic coordinate procedure does not appear to experience the same kind of "jamming" near boundaries that occurs with the gradient search. Thus, the cyclic coordinate search is another alternative to the fully-modified gradient search.

Our analysis leads us to believe that factors C (FREQUENCY SPACING), E (STARTING FREQUENCIES), and H (RUN LENGTH) do not have any great impact on the results of synthesis runs for our example problem. All of our results actually apply to our example problem only. The factors and interactions which we found to be significant would probably be significant for other synthesis examples as well. Still some caution is in order before these results can be extrapolated to arbitrary synthesis or optimization problems. For example, we might have found factors C and E to be significant if we had assumed a narrower available frequency spectrum.

Finally, we could use the information in Table 5 about the average effects to construct a predictive model to estimate the worst aggregate C/I ratio at any test point for the synthesis runs we did not make because of our fractional factorial design. If we let A, B, D, F, and G represent the five factors which produced the significant effects in our experiment and we let a value of -1(+1) stand for the low (high) level of each factor, the following equation can be used to predict the worst C/I ratio for any run we may be interested in:

$$\text{Worst C/I} = 30.68 + 10.95A - 2.29B + 3.98F + 2.13G + 2.29AB - 2.30BD + 2.67ABD \tag{7}$$

In addition to estimating the result to be obtained from a proposed run, this equation could be used to select promising runs if all possible specifications of values of A, B, D, F, and G are evaluated in (7). It does turn out that none of the predicted worst C/I values is over 50 dB. Three of the runs included in our experiment produced smallest C/I values in excess of 51 dB. Thus, we are lead to believe there is little reason to make any additional runs since we have already found solutions of such unmistakably good quality.

V. Conclusions

We have conducted an experiment to evaluate the usefulness of a gradient search algorithm and a cyclic coordinate search algorithm as tools for BSS synthesis problems. We have learned, first of all, that the problem as we have formulated it is a very difficult optimization problem. Both of these search techniques are known to perform well when the function to be optimized has an unconstrained unique optimum. This is not nearly the case with our problem. We see that by using different algorithms and different configurations for the starting solution we find markedly different solutions. The quality of the solutions found, as measured by the worst aggregate C/I ratio at any test point, also varies greatly.

We designed our 1/4-fractional factorial experiment so we could discover which variables, in addition to algorithm, affect the quality of our solutions. We have found that the separation between the satellites in the initial solution, the initial locations of the satellites, the length of the available orbital arc, and the width of the frequency spectrum also affect the quality of the solutions we found, either alone or in combination with other factors. A model for predicting the quality of the final solution for an arbitrary run was also constructed.

Because of the computational intensity of the evaluation of our objective function and the number of times it is evaluated, the cyclic coordinate algorithm does require a substantial investment in computing resources. The consistently good results with this method makes such an investment seem less risky. It may be that large synthesis problems can be solved by incorporating some sort of decomposition scheme into the solution procedure in order to reduce the total time required to find a solution. At this time, it is not obvious to us how this can be accomplished.

We believe that the greatest value of our experiment with BSS synthesis techniques is due to the likely applicability of these techniques to Fixed Satellite Services (FSS) synthesis problems. We feel that very similar results would have been obtained if we had concerned ourselves with an FSS synthesis test problem. The cyclic coordinate algorithm or a fully-modified gradient search may be an alternative to the nonlinear programming approach suggested by Ito et al. [3] as a means for solving FSS synthesis problems.

REFERENCES

[1] Anderson, V. and R. McLean, Design of Experiments - A Realistic Approach; Marcel Dekker, Inc., New York, 1974.

[2] Diamond, W., Practical Experiment Designs for Engineers and Scientists, Lifetime Learning Publications, Belmont, California, 1981.

[3] Ito, Y., T. Mizuno, and T. Muratani, "Effective Utilization of Geostationary Orbit Through Optimization" IEEE Transactions on Communications, vol. COM -27, No. 10, pp. 1551-58, October 1979.

[4] Levis, C., C. Martin, D. Gonsalvez, and C. Wang, "Engineering Calculations for Communications Satellite Systems Planning", ElectroScience Laboratory The Ohio State University, Columbus, Ohio, June 1983.

[5] Luenberger, D., Linear and Nonlinear Programming (Second Edition), Addison Wesley Publishing Co., Reading, Massachusets, 1984.

[6] Martin, C., D. Gonsalvez, C. Levis, and C. Wang, "Engineering Calculations for Communications Satellite Systems Planning", ElectroScience Laboratory, The Ohio State University, Columbus, Ohio, December 1983.