August 1974

Information Report FF-X-64

AIRPRO

An Air Tanker Productivity Computer Simulation Model

The Fortran Program

(Summmary)

by

A. J. Simard, A. Young, R. Redmond

Forest Fire Research Institute
Department of Fisheries and the Environment
Canadian Forestry Service
240 Bank Street
Ottawa, Ontario
K1G 3Z6

CONTENTS

AIRPRO

An Air Tanker Productivity Computer Simulation Model

The Fortran Program

(Summary)

A. J. Simard, A. Young, R. Redmond

This report summarizes AIRPRO. Its purpose is to give the reader a general impression of what the program does. Detailed documentation is not being generally distributed because 1) the cost of printing a large number of copies would be prohibitive, and 2) it is anticipated that only a limited number of individuals would be interested in a detailed description of the workings of the program. For those interested in more information than is included here, complete documentation is available from the Forest Fire Research Institute[1].

THE MODEL

Purpose

AIRPRO determines the optimum combination of air tanker resources and tactics to employ in individual wildland fire suppression operations. It accomplishes this by first simulating the fire suppression operation with unaided ground forces. In addition, it also simulates fire growth and the response of fire behaviour to changes in environmental conditions. It then fights the same fire employing a variety of air tanker resources and tactics. There are no predetermined dispatch rules. Every combination which could possibly reduce the cost-plus-loss is tested. It stores all results for subsequent analysis and lists the optimum combination based on the minimization of cost-plus-loss.

Finally, it accumulates individual fire results to produce overall system totals. Each fire is considered independently -- no provision is made for simultaneous fires or insufficient resources. While these are significant considerations, it has been decided to consider these problems separately, using data generated by AIRPRO as input.

---

[1] Information report FF-X-64 (Documentation).

AIRPRO has sufficient built-in flexibility to: consider from one to any number of fires; simulate from one to a large number of air tanker combinations; and generate a very limited or large amount and variety of output. The choice is dependent on options selected by the user. The primary purpose of AIRPRO is to anlyse the utilization component of an air tanker system within the context of the overall fire management system. There are several general uses for the data generated by the program.

1. Determine air tanker productivity on individual fires.

2. Examine the response of the air tanker system to changes in its operating environment.

3. Compare individual resources and tactics and identify the conditions under which each is superior.

4. Identify conditions under which air tanker resources should or should not be dispatched.

5. Examine the temporal and spatial variability of air tanker operations.

There are a number of practical uses for data generated by AIRPRO.

1. Aid in selecting the type and amount of air tanker resources best suited to and required in an administrative area.

2. Develop individual fire preattack dispatching guides.

3. Aid the dispatcher on a real-time basis for individual or multiple fire situations (with modifications to the current program).

4. Determine the best allocation of resources in cases where only a few combinations are being considered or to generate input data to be used by an allocation algorithm, in cases where many combinations are being considered. (As long as the individual fire assumption does not significantly detract from the validity of the results).

Language

The program is written in Fortran. It is currently operational on a Univac 1108 computer, controlled by a "General Programming System" (GPS) developed by Computer Sciences Corporation.

A few systems specific functions are employed; a call to the internal clock and some mathematical functions. These were kept to a minimum to make it as easy as possible to modify the program to fit other systems.

Size

AIRPRO contains just under 3500 lines of programming. There are approximately 750 variables. The length of individual subroutines varies from 70 to 500 lines. The number of variables in the subroutines varies from 8 to 108, with 190 variables in common blocks. Computer storage requirements are 20 K words for the program and 10 K words for data, making a total of 30 K words. Since the GPS system uses a 4 byte word, the above translates to a total of roughly 120 bytes. It is felt that the storage requirements could be halved with overlaying. In general, a medium to large size computer is required to run the program.

Operation

On the system being used, accounting information is given in terms of System Resource Units (SRU's) which integrate time and hardware utilization. Therefore, it is not possible to determine the time required to compile and link the program. Compilation on the Fortran IV compiler required an average of 83 SRU's and linkage 11. On a time sharing basis, the cost of these two operations would amount to $19.00. On a low priority batch basis, the cost would be reduced to $4.25.

Total run time under Fortran IV for a sample of 3,010 fires (on which 53,655 combinations were analysed) was 2,169 seconds (36.1 minutes). This averaged out to 0.72 seconds per fire -- a relatively short time, considering the considerable amount of computation involved. In terms of cost, 9,767 SRU's were used, for a total cost (at minimal priority) of $439.50. Of the above total, program set up and tabulation requires about 12 SRU's or 0.1 percent. This is roughly equivalent to the average cost of simulating three fires.

File Organization

There are a number of files used and generated by AIRPRO. Each is discussed below. Sample listings of each file can be found in the program list section.

1. Job Files - these are short files with a series of job and system commands which instruct the system to establish the necessary files and execute the program. Two files were used.

- LINKGO - for programming and debugging. This version contains implanted debug commands. It was designed for one or a few fires and provided detailed output as required.

- GOTAPE - for volume testing and production runs, using tape files only.

2. Program Files - each subroutine is stored in a separate file with the subroutine name being used as the file name. There are three types of program files.

   - Subroutine - PNC: the source language file (1 per subroutine)
   - Subroutine - REL: the relocatable compiled subroutine (1 per subroutine)
   - MAIN - XQT: the executable program (1 only)

   The source language program files will be discussed in detail subsequently.

3. Input Files - there are two input files required by the program:

   - DATA1 - a disk file, containing all the fire management system data. It is accessed once at the beginning of the program. The total length of DATA1 is variable, depending on the number of air tankers to be tested. The current version contains the equivalent of 370 cards.

   - FIRES_ - a tape file containing individual fire data. From one to any number of fires can be simulated. FIRES7 contains 3,010 fires from the province of New Brunswick. FIRES1 through FIRES6 were earlier versions of the data to which editing and other modifications were made. Each fire record contains 305 bytes of data and is stored as 3 lines to facilitate listing. Total current file length is 9,030 lines of 105 characters each. A disk file - FIRES was created to contain various samples of from 1 to 300 fires for testing and debugging.

4. Output Files - in addition to a variety of terminal or printer output as selected by the user, the program can generate three output files, while a fourth is generated by a supplemental program.

   - INFO - a disk file containing an echo check of the system data, including table headings, to facilitate reading by nonprogramers.

- EVERY_ - a tape file containing the results of every simulation completed by the program. EVERY7 (the current version) contains 59,810 lines (1 per trial) not exceeding 132 characters in length. A disk file - EVERY - was created to contain output from early runs.

- BEST_ - a tape file containing all the information found in FIRES7 plus the optimal result for each fire. The results are contained on one line, if ground suppression was best and two lines if air tanker suppression was best. BEST7 (the current version) contains 12,789 lines, not exceeding 132 characters each. A disk file - BEST - was created to contain output from early runs.

- SNOW - a tape file generated by MODEL (a supplementary program). It is in the same format as BEST. It contains the optimal result for every fire using only the S2D Snow Commander. Fire and ground suppression data are omitted for those fires where the Snow Commander was not tested.

OUTLINE

The program consists of a MAIN, 17 subroutines and 34 entry points. The overall function of each component is outlined in the remainder of this section. To aid in cross referencing with the model, the subroutines are grouped by model component with an additional group for those subroutines which are primarily administrative. Note that for the sake of programming efficiency, some subroutines perform functions pertaining to more than one model component.

I. Administration

MAIN

MAIN controls the program. It calls administrative routines which bring in data, tabulate and output results. By means of an event calendar, it determines the sequence of calling model subroutines.

1. Subroutine DATAIN

   DATAIN brings in all system and environmental data needed by the program.

2. Subroutine RESULT

   RESULT calculates and stores savings and outputs the results of each tactic. It also summarizes and outputs the results for each fire and the overall run.

- Entry INTAB

  Initializes tabulation variables for the overall run.

- Entry RITAB

  Reinitializes tabulation variables at the start of a new fire.

II. Environment

3. Subroutine GEO

   GEO calculates the distance between pairs of locations related to the fire suppression operation.

4. Subroutine SUNSET

   SUNSET calculates the time of sunrise and sunset.

5. Subroutine UPDATE

   UPDATE revises environmental and fire behaviour parameters on an hourly basis.

   - Entry CYCLE

     Calculates adjustment needed to change observed average spread rate over time to initial rate.

III. FIRE

6. Subroutine FIREG

   FIREG controls the growth of the fire.

   - Entry ENFIRE

     Calculates total fire perimeter and area when suppression is finished.

   - Entry NEWRS

     Adjusts growth rate on new flank when old flank is contained.

   - Entry NEWRS2

     Adjusts growth rate on 2 adjacent flanks when suppression starts.

- Entry RSARC

  Resets arc growth variables to the time of control of the last flank by air tankers.

- Entry STOARC

  Stores arc growth variables at the time of control of each flank by air tankers.

7. Subroutine FIREIN

   FIREIN brings in all data related to a specific fire. It also edits and modifies the data as required by the program.

   - Entry ESCAPE

     Lists fire number for escaped fires.

8. Subroutine INFIRE

   INFIRE initializes all fire related variables.

   - Entry AVGINT

     Calculates average intensity for retardant effectiveness, based on both free burning and suppression growth rates.

   - Entry INARC

     Gets initial fire data needed for preliminary tests.

   - Entry INFIRE2

     Adjusts fire behaviour variables to reflect observed suppression growth rate and estimated time to control.

   - Entry INTMAX

     Calculates maximum possible intensity, based on fuel type, size class and season.

   - Entry RIFIRE

     Reinitializes fire parameters to the start of suppression.

   - Entry RSFIRE

Resets fire parameters to the time of control of the last flank by air tankers.

- Entry SFIRE

Stores fire parameters at the time of control of each flank by air tankers.

IV. Ground

9. Subroutine ECON

ECON calculates economic variables related to the ground and air tanker systems.

- Entry CAIR

Calculates the cost of the air tanker operation.

- Entry CGND

Calculates ground suppression cost.

- Entry DAMAGE

Calculates damage.

10. Subroutine GNDSUP

GNDSUP controls the fire using ground forces.

- Entry CREWS

Initializes ground suppression variables when ground forces arrive.

11. Subroutine INGND

INGND initializes the ground suppression system. It also stores and reinitializes parameters when new tactics are selected.

- Entry RATE

Adjusts rate of line construction for arrival of a new crew.

- Entry RIGND

Reinitializes the ground system to the start of the fire.

- Entry RSGND

Resets ground system to time of control of the last flank by air tankers.

- Entry SGND

Stores the ground system when air tankers control a flank.

12. Subroutine MOPUP

MOPUP calculates the time required for mop-up.

V. Air

13. Subroutine AIRSUP

AIRSUP fights the fire with air tankers.

14. Subroutine DELVRY

DELVRY calculates all parameters related to the delivery subsystem.

- Entry CIRCUT

Calculates circuit times for each air tanker and type of operation.

- Entry RSTIME

Changes water-based flying times to land-based and vice versa.

- Entry TNDROP

Calculates the time of the next drop and bird dog flying time.

15. Subroutine DROP

DROP calculates all parameters related to the drop subsystem. It also calculates the depth of retardant required.

16. Subroutine INAIR

INAIR initializes the air tanker system. It also stores and reinitializes parameters when new tactics are selected.

- Entry NEWARK

  Finds the next flank of attack for air tankers.

- Entry RIAIR

  Reinitializes the air tanker system to the start of the fire.

- Entry RSAIR

  Resets air tanker system variables to the time of control of the last flank by air tankers.

- Entry SAIR

  Stores air tanker system parameters when a flank is controlled by air tankers.

17. Subroutine SELECT

    This subroutine selects, in sequence, various combinations of air tanker resources and tactics for analysis by the simulation model. It also performs simple preliminary checks to determine whether or not it is worthwhile to test specific combinations.

    - Entry INSEL

      Determines the order of selection of primary air tanker models.

MAIN

A general impression of the functioning of the program can be obtained by examining MAIN. As can be seen in the accompanying flow diagram, MAIN contains four loops. The outermost fire loop brings in new fires and initializes the system. The tactic loop controls the selection of tactics and resets the system to the appropriate state. The event loop is the heart of the program, controlling the calling of most of the technical subroutines. The innermost loop controls fire growth and ground suppression. In addition, MAIN controls program initialization and termination. The following discussion describes MAIN in more detail.

The first function is to initialize the program. Three activities are involved: 1) bring in system data, 2) initialize program tabulation variables, and 3) initialize the air tanker system. MAIN then enters the outermost fire loop and brings in data for one fire. All fire and some ground suppression parameters are then initialized. This is followed by a calculation of the area, control time, suppression cost and losses that would be incurred, if the fire could be immediately controlled. Finally, the first four elements of the event calendar are initialized by calculating the elapsed time from detection to the first occurrence of each.

The program then enters the tactic selection loop. If the tactic is a new location of attack, fire and system parameters are reset to the time of completion of the last flank by air tankers. This process avoids having to repeat the simulation up to that point, resulting in a significant saving in computation. For all other tactics, the fire and system parameters are reset to the time of detection.

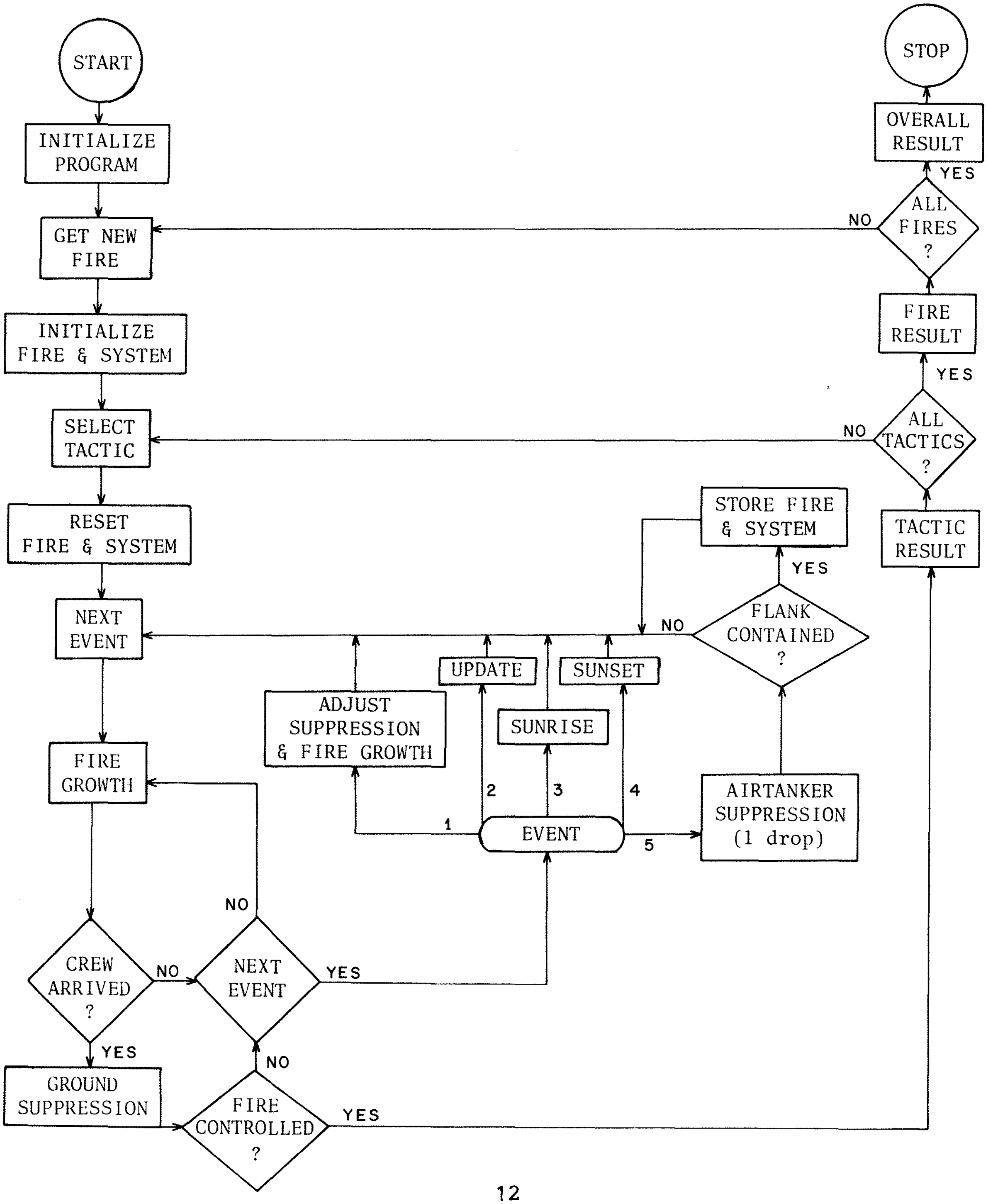
MAIN then enters the event loop to run the simulation using the selected tactics on the current fire. First, the next event is identified by searching for the smallest elapsed time in the event calendar. The program then enters a small fire growth and suppression loop. The purpose of this loop is to set a maximum time limit between calls to fire growth and suppression. If the interval between events is less than 20 minutes, only one call is made. If the interval is longer (there is a maximum of one hour between calls to UPDATE) up to three calls can be made during the interval.

After fire growth and suppression are completed, one of five events take place (crew arrival, update, sunrise, sunset or an air tanker drop). Appropriate adjustments are made to fire and system variables, depending on the event. In addition, the time of the next occurrence of the current event is calculated. If an air tanker drop contains the flank currently being attacked, the state of the fire and system, at the time the next drop is due, is stored for future resetting. The program then finishes controlling the fire with ground forces.

When the entire perimeter is controlled by ground forces, the program enters the result sequence. First, final acreage as well as control and mop-up times are determined, followed by costs and losses. Control then returns to SELECT where current results are compared with previous values. If another tactic is to be tested on the same fire, MAIN reenters the tactic loop. If not, the optimum combination for the fire is determined and overall results are tabulated. Then, if there are more fires to be tested the program reenters the fire loop. If not, the overall results are listed, and execution terminates.

MAIN

SIMPLIFIED FLOW DIAGRAM



SUPPLEMENTAL AND TABULATION PROGRAMS

From the beginning, it was realized that it would not be possible to anticipate all possible uses to which AIRPRO would eventually be put. Further, while the cost of a run for a few fires would be inconsequential, the cost of a production run for a large data set necessitates that a minimal number of the latter type of runs be carried out. To rerun the program for each new tabulation that was desired would be prohibitively expensive. Further, integrating the supplementary programs into AIRPRO would have increased the storage requirements by a factor of 3 -- clearly an undesirable and potentially fatal side effect. It was therefore decided that AIRPRO would produce only a basic and limited output during the production run. In the case of real time analysis of a few fires, the production run diagnostic output is all that is needed, as specific solutions are the primary interest. In the case of runs on a large set of historical fires, however, additional summary tabulations are needed. All detailed summaries and tabulations are produced by separate programs that use the two major output files BEST and EVERY as input. These programs are outlined in this section.

Ten supplemental programs have been written and run. These programs generate a comprehensive set of summaries and tabulations. They provide a wide range of information which can be used for many different types of analysis. The programs are listed below. A variety of statistics are shown in Table 1.

1. FREQ1 - Produces frequency distributions and averages for a series of fire, ground and environmental variables associated with each resource and tactic when it was optimal.

2. FREQ2 - Produces frequency distributions for fire locations associated with each resource and tactic when it was optimal.

3. GNDTST - Outputs selected fire and suppression data when simulated results, with ground forces only, differ significantly from observed results. Also outputs frequency distributions of simulated and actual results and summarizes differences.

4. MAXSAV - Summarizes savings and production totals by air tanker model and number of air tankers.

5. MODEL - Generates a file, in the same format as BEST, for one air tanker model.

6. PATERN - Calculates maximum pattern lengths for various retardant depths and tank combinations.

7. SIMFIR - Estimates the number of simultaneously occurring air tanker fires.

8. SUBCAL - Compares simulated versus observed ground suppression results for several sample sizes.

9. SUBOPT - Compares savings of selected optimal resources and tactics with savings for non optimal combinations.

10. SUBTOT - Summarizes production and savings totals by: 1) fire season, 2) air base and 3) forest.

TABLE 1.

SUMMARY OF STATISTICS FOR SUPPLEMENTAL PROGRAMS

Program Name	Lines	Storage Required (Words)	Execution Time (Sec.)	Cost
FREQ1	263	36K	43	$ 8.75
FREQ2	232	35K	43	$ 8.75
GNDTST	287	8K	26	$ 5.50
MAXSAV	286	16K	185	$37.50
MODEL	82	8K	230	$46.50
PATERN	148	7K	3	$ .75
SIMFIR	113	7K	2	$ .60
SUBCAL	168	7K	23	$ 4.75
SUBOPT	208	10K	140	$29.00
SUBTOT	225	16K	30	$ 6.00