N89-10067

AUTOPLAN - A PC-BASED AUTOMATED MISSION PLANNING TOOL

Frank C. Paterra
Marc S. Allen

Computer Technology Associates, Inc.
Newport News, VA 23606

George F. Lawrence

Langley Research Center
Hampton, VA 23665

ABSTRACT

The Evolutionary Definition Office (EDO) at the Langley Research Center (LaRC) has the responsibility to analyze and evaluate alternative growth options of the Space Station and its utilization. Under contract to the EDO, Computer Technology Associates (CTA) has developed a PC-based automated mission and resource planning tool, AUTOPLAN. AUTOPLAN's input is a proposed profile of missions, including for each: start year, number of allowable slip periods, mission duration, and requirement profiles for one or more resources as a function of time. The user also inputs a corresponding availability profile for each resource over the whole time interval under study. Subject to the size of a given problem and microcomputer performance limitations, AUTOPLAN finds all integrated schedules which do not require more than the available resources.

AUTOPLAN is implemented in Arity compiled PROLOG, and executes on an IBM PC/AT with 640 KB memory. There is particular interest in small-scale planning and scheduling systems in the Space Station program because of the trend toward decentralizing these functions. The iterative resolution and recursion features of PROLOG greatly simplify the programming of this problem, and make it easy to customize or generalize the solution evaluation algorithm. The quantitative capabilities of the tool and several postprocessor interpretive aids presently under assessment are described, and a realistic sample application of the tool suite is presented.

1. Introduction

The Space Station Mission Requirements Data Base (MRDB) contains information on over 300 scientific, technology development, and commercial missions proposed for the Space Station. Each of these missions is characterized by requirements for supporting resources, including crew time for assembly, servicing, and operation, electrical power, thermal dissipation, and communications. The design of the flight segment must be able to allocate these resources among the various payloads installed at any given time. The implication of this allocation is a need for comprehensive resource scheduling and operations coordination.

With Phase B preliminary design studies complete, enough is known about the configuration and build-up of the Space Station to allow meaningful comparisons between the projected demand for resources and the availability of these resources over time. Recent studies have shown that many resource categories will be seriously oversubscribed from the outset of Station activation. A recent McDonnell Douglas Astronautics Company (MDAC) study for the EDO has shown that many of the more resource-intensive payloads will probably have to be operated intermittently if all customers are to be accommodated. Resource scheduling and operations coordination are thus developing into major aspects of integrated Station operations.

Because of the great complexity of the Station system, efficient distribution of available resources will have to be automated to a high degree. The competition of a large number of host and user subsystems, with widely differing requirements and operating priorities and options, for many dissimilar resources will not be adjudicable by manual methods. Worse, an important design objective supporting operational flexibility is replanning on as short a time scale as possible.

These factors suggest that the scheduling and coordination approach likely to be adopted will be a hierarchical one, where gross allocations will be made in a top-down fashion, and detailed schedules will be propagated upwards for integration into master schedules. Naturally, there will be a degree of iteration in this flow. A general approach of this type has the most promise for supporting a telescience operations concept and allowing maximum latitude in detailed planning and operations to end users.

One beneficial effect of a hierarchical approach to resource allocation and scheduling is that the problem is more tractable at each level than a simultaneous, global solution would be. This fits in well with a user accommodation concept emphasizing distributed operations; it also suggests the need for automated planning tools suitable for end-users. AUTOPLAN can be viewed as a prototype for an automated scheduling tool which is capable of solving complex scheduling problems on a modest hardware configuration likely to be found at any end-user site.

2. AUTOPLAN Algorithm

The algorithm used by AUTOPLAN to search for successful mission sets must take into account the large search space that could be involved. In order to reduce that search space, AUTOPLAN searches for solutions in a recursive-descent, tree-like manner. Each mission is a node in this tree, and the number of branches at that node is the number of slips allowed for that mission. Successive missions on the list are then extension sub-branches. The central looping core of AUTOPLAN is a mixture of both iteration and recursion. A mission is appended to the solution set by iteratively reading and testing successive entries for that mission. If one of these tests is successful, the next mission on the mission list, and thus the next level (node) of the search tree, is tried through recursion. As AUTOPLAN moves down the search tree, it maintains a running sum of resource use of each resource for each time period and compares them against the corresponding resource envelopes; if the partial sum of any resource for any operating period becomes larger than the corresponding available resource, that search path is truncated. This allows AUTOPLAN to discard many failure paths without examining them because the tree is cut at that node and all subsequent branches are summarily removed from the search space. The result of this truncation is a significant improvement in performance for problems with sparse solutions.

Because AUTOPLAN maintains these running sums, and execution time will be reduced if branches close to the root are cut off, an analyst can speed up the searching process by arranging the missions in deceasing resource use order. In addition, since AUTOPLAN saves failure as well as solution sets, an analyst can order the missions by priority, and then examine the failure file for partial mission sets which were able to accommodate at least the high priority missions.

3. AUTOPLAN Implementation

AUTOPLAN was written in compiled Arity PROLOG. Because of the backtracking and unification features of PROLOG and its natural support for list processing and recursion, the tree-oriented search was easily implemented. During design and implementation, it was recognized that the searching algorithm should be optimized for speed. To achieve this, interpretation and display of solutions and failures have been isolated from the searching algorithm. As a result, the raw output is not very readable without postprocessing. Also, there is no error checking of mission data during the solution search so all data must be entered through a preprocessor. To retain maximum flexibility, AUTOPLAN was developed with no resource or time period limits, other than the computing power of the host computer. Any number of resources and any time period granularity can be accommodated.

To run AUTOPLAN, an input data file is created from mission and resource data using a data set editor. As the program is executed, the input data file is read, and each mission with allowed slips is expanded. Expanding a mission consists of creating an additional entry for the mission for each time period it is allowed to slip. Once all missions have been read and expanded, AUTOPLAN asks the user whether a single solution or all solutions are desired, and then prompts for the type of runtime display. There are three types of runtime displays: Full Graphics, Resource Use Only, or Successes and Failures Only.

Full Graphics Display - This option displays a grid that highlights periods with excessive requirements, a dynamic list for each resource that shows the actual quantity being used in each period, and counts of successes and failures (Figure 2).

Resource Use Only - This option displays a list for each resource being examined and the counts of successes and failures.

Successes and Failures Only - This option displays only the success and failure counts tried so far.

Lastly, the user is prompted for the solution and failure file names, and the search begins. Full solutions and partial lists for failed permutations are saved in these files for input to the postprocessor. Once the last solution is found, the elapsed time is displayed and the program ends. The solutions and failures are viewed using postprocessing software.

4. <u>The AUTOPOST Postprocessor</u>

When AUTOPLAN identifies a solution or a failure, it writes it to the solution or failure file. The solution and failure records in their raw form are not very useful to a human analyst, because they are in a highly compacted, PROLOG-readable form. The postprocessor reads the solution and failure files produced by AUTOPLAN and presents the data to the analyst in a more intelligible form.

The present postprocessor package, AUTOPOST, offers three postprocessors:

Mission Schedules - This produces a chart, similar to Figure 3, graphically showing the operational interval of each mission in a solution set. A separate chart is displayed for each of the solution or failures sets.

Average Resource Use - This produces a histogram for each of the resources showing the average amount, averaged over all successful schedules, consumed in each period. The envelope of available resources is also shown on the report (Figure 4).

Most Efficient Solution - Because by definition all missions must be flown in each schedule solution, the integrated consumption of any resource for all solutions is the same. This postprocessor searches the solution file for the solution whose peak usage of a selected resource is the smallest. In the following example, solution 2 would be selected as the "most efficient" solution.

Solution 1	Solution 2
[10,11,12,9,6,10]	[10,8,11,9,10,10]

Because of the modularity of AUTOPOST's design, additional postprocessors can be added easily as their need is identified by Space Station analysts.

5. Case Study - Technology Development Attached Payloads

In order to conduct an illustrative, yet realistic, analysis with the available data, CTA selected 14 technology development missions, both U.S. and foreign, from the list of 33 attached payload missions grouped together in a recent MDAC evolution study. Technology development missions were selected because of their special interest to the LaRC Space Station Technology Office.

Although AUTOPLAN's full graphics display is capable of handling up to five resource categories at a time, and the central algorithm has no limitation at all, only power and IVA crew time for daily operations were analyzed in this study. Other interesting parameters available in the MRDB, such as physical volume and up- and downmass, need additional information before they can be applied as quantitative schedule constraints.

2.1 ASSUMED MISSION LIST

The mission subset chosen for study consists of Manned Base "attached payloads" not included in any of the other MDAC categories, further qualified by being either U.S. or foreign technology development missions. All SAAX missions, therefore, were excluded, as were all Japanese S-XXX and E-XXX missions.

Some of the 14 missions on the resulting list showed essentially continuous resource requirements after activation; others tended to operate for a year or so at a time, skipping one or more years between operational periods. In defining allowable slips for the missions on the list, the following rule was applied: for missions with continuous operation, allow no slips unless the first operational year is 1994, when resources are especially scarce; for missions beginning in 1994 or having embedded non-operational periods in the schedule, allow one or more slip years.

This principle does not represent any programmatic considerations, but it is conducive to optimum use of resources.

Although the missions in the list are baselined in the MRDB against a 1992 start date, their schedules are all mapped here onto a LaRC Critical Evaluation Task Force (CETF) 1994 resources timeline. This is equivalent to an a priori, uniform two year delay for all missions. Individual slips for selected missions are then applied to this modified schedule, as described above.

The resulting mission list, with assigned allowable slips, is as follows:

Mission	Allowed Slips	Mission	Allowed Slips
TDMX2441	0 years	TDMX2321	1 years
TDMX2011	0	TDMX2574	0
TDMX2132	0	T-007	0
T-001	1	TDMX2542	1
TDMX2061	1	T-008	0
TDMX2153	2	TDMX2541	2
TDMX2311	1	TDMX2543	0

The total number of possible schedule permutations for these input data is:

$$288 = 1^7 \times 2^5 \times 3^2$$

5.2 ASSUMED INDIVIDUAL MISSION RESOURCE REQUIREMENTS

This study considers only payload requirements for power and daily IVA crew time. Crew time requirements for setup, servicing, reconfiguration, and teardown are not addressed. The data used were extracted from the EDO MRDB (EDOM), using a partially implemented data extraction program.

In order to compute daily requirements for these two resources, it is necessary to interpret the contents of the data base. For power, a conservative algorithm is applied: if a given payload is operational on any day in a given year, it is assumed to draw resources every day of the year. Standby, normal operating, and peak demands are combined in proportion to their time fractions as tabulated in the data base. For crew time, a more liberal algorithm is applied: it is assumed that crew needs can be scheduled against one another within each year period in such a way as to minimize conflicts. That is, the mean daily crew time for a given payload over a year is computed by prorating the requirement per operational day by the number of days in that year that the payload was in operation. It should be noted that either algorithm, i.e., the conservative or liberal one, could be applied to any resource. Or, an average requirement could be computed from the two algorithms. The choice amounts to an emulation of the results of scheduling at a more detailed level.

Figure 1 presents the power and crew data extracted from the EDOM, as shown in a report produced by the EDOM Mission Analysis Tool (EMAT) software.

5.3 ASSUMED TOTAL USER RESOURCE ENVELOPES

The CETF briefing presents profiles for total user allocations of both power and IVA crew time; the resources provided to attached payloads must be a subset of this total user allocation, and technology development attached payloads will, in turn, be allowed a portion of this subset.

For power, a CETF graph shows approximately 20 KW available to all users until the solar dynamic generating system is installed in the last quarter of 1994. Then the user power resource increases to about 70 KW, from which it gradually declines because of mounting system requirements for the growing Station to about 60 KW in 1997. For the first year, 1994, the average total user power is then:

$$32.5 \text{ KW} = 0.75 \times 20 \text{ KW} + 0.25 \times 70 \text{ KW}$$

Note that this method of resource combining is only a compromise between conservative and liberal approximations, and does not strictly represent detailed scheduling within the one year time period granularity. For example, this computation suggests that a solitary 25 KW device could be operated all year, whereas it actually could be operated for only the last three months. Similarly, it incorrectly suggests that a single 60 KW experiment could not be operated at all. This inaccuracy in accounting for scheduling within the finest time granularity can be reduced by using finer time divisions, but it will also be reduced for cases with less abrupt changes in resource availability or for situations where the most demanding resource sinks absorb a smaller fraction of the total available.

For 1995, the average total user power is 70 KW. It is assumed that there is a linear decrease for the next two years to 60 KW, after which, in the absence of additional information, total available user power is assumed constant. Further increases in system requirements might be offset by improved efficiency, addition of capacity or other augmentations. This leads to the following profile for total user power, expressed in KWhour/day:

Year	KW	KWhour/day
1994	32.5	780
1995	70	1680
1996	65	1560
1997-2003	60	1440

A similar argument is adopted for IVA crew time. From the CETF briefing, there is no permanent crew presence until the final third of 1994. From then through the first third of 1995, the graph indicates about 72 hours per week available to all users. From then until the end of the first third of 1996, the total weekly allocation is 128 hours. Finally, after that point, 244 hours per week of IVA crew time are available for sharing by all users. The necessary computation for 1994 is:

$$24.0 = 0.33 \times 72 \text{ hours}$$

For 1995:

$$109.3 = 0.67 \times 128 \text{ hours} + 0.33 \times 72 \text{ hours}$$

For 1996:

$$205.3 = 0.67 \times 244 \text{ hours} + 0.33 \times 128 \text{ hours}$$

For 1997 and later years, the answer is simply 244 hours per week of IVA crew time for all users. Converted to IVA crew hours per day for all users, this is:

Year	Hours/week	Hours/day
1994	24.0	3.4
1995	109.3	15.6
1996	205.2	29.3
1997-2003	244.0	34.9

5.4 ASSUMED TECHNOLOGY DEVELOPMENT RESOURCE ALLOCATION

As hypothesized above, the technology development attached payload user community can expect to be allocated only a fraction of the total resource pool available to all users. The precise fraction must be set on policy grounds. Other user groups include science and commercial users, and, within each of these groups, attached payloads must compete with laboratory equipment and servicing requirements. This case study assumes that the technology attached payload community will be allocated 20%. The resulting distributions for power and crew IVA time are listed in the following tables:

YEAR	POWER (KWhour/day)	CREW (IVA Manhour/day)
1994	156	0.7
1995	336	3.1
1996	312	5.9
1997-2003	288	7.0

It is evident that IVA crew time is an especially scarce resource, especially in the first two years of manned operations. The crew requirement analyzed here includes only regular periodic operations, and omits IVA needs for setup, configuration changes, servicing, and teardown. According to Figure 1, five of the missions on the list do not require any of this periodic crew activity.

5.5 RESULTS

Performance figures given in the discussion are based on execution on a PC/AT with 1.1 MB RAM; 512 KB of this RAM are configured as RAM-disk containing the input data set and PROLOG's dynamic data base. Additional, but small, performance improvement is possible by writing the output success and failure data sets also to RAM-disk.

For the resource availability assumptions used, the number of failures found is 95, with only 4 possible solutions out of 288 possible permutations. AUTOPOST displays of the solutions are shown in Figures 3 and 4. Figure 3 shows the feasible mission timeline for each solution, and Figure 4 shows the total resource consumption as a function of time, averaged over all missions. This second plot, which also lists and displays the limiting resource envelope, is helpful in gaining a qualitative understanding of the critical resources and critical times. Execution time required was 2 minutes and 3 seconds.

Two additional cases were run to observe AUTOPLAN's behavior in discarding solutions for this data set as the envelopes were shrunken. The algorithm concluded that no solution was possible if only 14% of the total resources are allocated to this mission set. This was determined after examining 77 profiles of the 288 in 1 minute and 23 seconds.

In the extreme case of only 12% allocation, the first examined mission with a non-zero crew requirement, TDMX2132, required 0.5 manhours per day. This could not be provided by the total crew allocation. Since this mission was not allowed any slippage, AUTOPLAN terminated its search with this single failure in less than a second.

5.6 CASE STUDY CONCLUSIONS

The results presented for this illustrative example allow a number of conclusions to be drawn about the assumed scenario for scheduling the technology development attached payload missions and about AUTOPLAN and its use.

- The feasibility of an integrated schedule is critically dependent on the resources available. A change of even a few percent can mean the difference between many choices in schedule and no solutions at all.

- Separated from other mission groups as done here, the assumed model of the technology development community would need approximately 20% of total power and daily IVA crew time available to users. This does not include requirements for setup, servicing, and teardown. Of course, it is not necessary that the percentage allocation of different resources (e.g., power and crew) be the same.

- Crew requirements for setup, servicing, and teardown should also be included in the over-all crew requirement evaluation.

- Contingency margins should be subtracted out of allocated envelopes.

- Policy guidance should be available to help assign slip allowances, if realistic results are to be obtained.

- The fidelity of the results are only as accurate as the input resource budgets and input mission requirements;

determining these are the limiting factors for study accuracy.

- AUTOPLAN exhibited adequate performance for a realistic problem on a widely available equipment configuration. Although the processing power of the PC/AT does limit the size of problems which can be solved at once, it is adequate to support useful analyses at individual levels in a hierarchical allocation model.

- AUTOPLAN enables the mission analyst to perform schedule evaluations not possible by other means.

- Postprocessor functions are the key to making the results useful; these functions need not be limited to simple displays, but could include additional logical or arithmetic operations, since the solution data set fully characterizes all solutions. One candidate is a precedence filter which could select all solutions wherein certain missions are completed before initiation of others.

- The code is not restricted to problems based on ten, one-year time intervals; that is, it could evaluate daily schedules over a month, or hourly schedules over a day. However, a flexible data set editor is required to simplify input data set construction for input data other than standard MRDB timelines. Development of such an editor is planned as a follow-on activity.

6. Future Plans

Although AUTOPLAN is capable of analyzing data from any source, its use is presently restricted by limited support tools to data extracted from the EDOM reorganized version of the MRDB. Several postprocessor functions are also in place, as illustrated in the case study. Enhancements to the tool suite can be roughly divided into three groups: extensions to the algorithm itself, improvements in input data set construction, and addition of postprocessing functions.

The basic functionality of AUTOPLAN's search and evaluation algorithm is very general. Most enhancements to the main program are likely to provide additions to reporting of search by-product information for postprocessor use, or alternatives to the evaluation algorithm. For example, the simple add-and-compare test for schedule viability could be replaced by a more sophisticated test. In general, however, increased complexity in the core processing will degrade performance in searching the candidate solution space. Wherever possible, enhancements should be implemented through pre- or postprocessor functions.

One of the major limitations of the present package is difficulty getting data into AUTOPLAN. No software is currently available to help a user edit input data extracted from the EDOM: mission parameters must be used exactly as contained in the data base, even if the user has better information. Secondly, although AUTOPLAN is capable of solving entirely different problems, for example, scheduling individual hours during the day or individual days during the month, no tool is available to construct input data sets from scratch. It is expected that capabilities in both of these areas will be developed during a follow-on task.

Finally, a diversity of postprocessors will be needed to support the needs of different analysts. The postprocessors implemented so far are simply the most obviously useful. Since AUTOPOST operates on only known solutions, brute performance is not the over-riding concern that it is for the AUTOPLAN search algorithm. As a logic programming environment, PROLOG facilitates the construction of complex logical inferencing functions. New modules could be implemented and integrated easily into the AUTOPOST framework. Use of the AUTOPLAN/AUTOPOST package on real Space Station problems is expected to suggest numerous useful extensions.

This work was performed under contract NAS1-18247.

FIGURE 1 - EDOM POWER AND CREW REQUIREMENTS

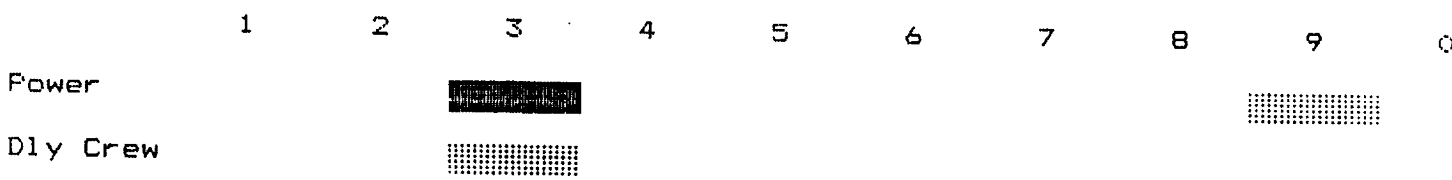
Table of power use (KWH/Opnl Day)

Mission	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001
TDMX2441	6.1	6.1	6.1	6.1	6.1	6.1	6.1	6.1	6.1	6.1
TDMX2011	11.2	11.2	11.2	11.2	11.2	11.2	11.2	11.2	11.2	11.2
TDMX2132	18.0	18.0	18.0	18.0	18.0	.0	.0	.0	.0	.0
T-001	24.0	24.0	24.0	24.0	24.0	24.0	24.0	24.0	24.0	.0
TDMX2061	6.0	6.0	.0	.0	.0	.0	.0	.0	.0	.0
TDMX2153	48.0	48.0	48.0	.0	.0	.0	.0	.0	.0	.0
TDMX2311	.0	24.0	24.0	.0	.0	.0	.0	.0	.0	.0
TDMX2321	.0	20.0	20.0	.0	20.0	.0	20.0	.0	.0	.0
TDMX2574	.0	.0	7.2	.0	.0	.0	.0	.0	.0	.0
T-007	.0	.0	.5	.0	.0	.0	.0	.0	.0	.0
TDMX2542	.0	.0	54.0	.0	54.0	.0	54.0	.0	54.0	.0
T-008	.0	.0	.0	.0	.0	7.2	7.2	.0	.0	.0
TDMX2541	.0	.0	170.0	.0	.0	170.0	.0	.0	170.0	.0
TDMX2543	.0	.0	.0	.0	.0	56.0	56.0	56.0	56.0	.0

Table of crew use (Avg hours/day)

Mission	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001
TDMX2441	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
TDMX2011	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
TDMX2132	.50	.50	.50	.50	.50	.00	.00	.00	.00	.00
T-001	.50	.50	.50	.50	.50	.50	.50	.50	.50	.00
TDMX2061	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
TDMX2153	.50	.50	.50	.00	.00	.00	.00	.00	.00	.00
TDMX2311	.00	.99	.99	.00	.00	.00	.00	.00	.00	.00
TDMX2321	.00	.10	.10	.00	.05	.00	.10	.00	.00	.00
TDMX2574	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
T-007	.00	.00	.11	.00	.00	.00	.00	.00	.00	.00
TDMX2542	.00	.00	.82	.00	.82	.00	.82	.00	.82	.00
T-008	.00	.00	.00	.00	.00	1.00	1.00	.00	.00	.00
TDMX2541	.00	.00	1.97	.00	.00	.66	.00	.00	.66	.00
TDMX2543	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00

FIGURE 2 - FULL GRAPHICS DISPLAY



[41.305,157.305,383.005,107.305,133.305,218.505,122.505,41.305,265.305,41.305]

[0.5,2.585,5.49,1.5,1.871,2.158,2.421,0.5,1.98,0.5]

Failures: 4

Starting time - 11:56:29

FIGURE 3 - SOLUTION TIMELINES

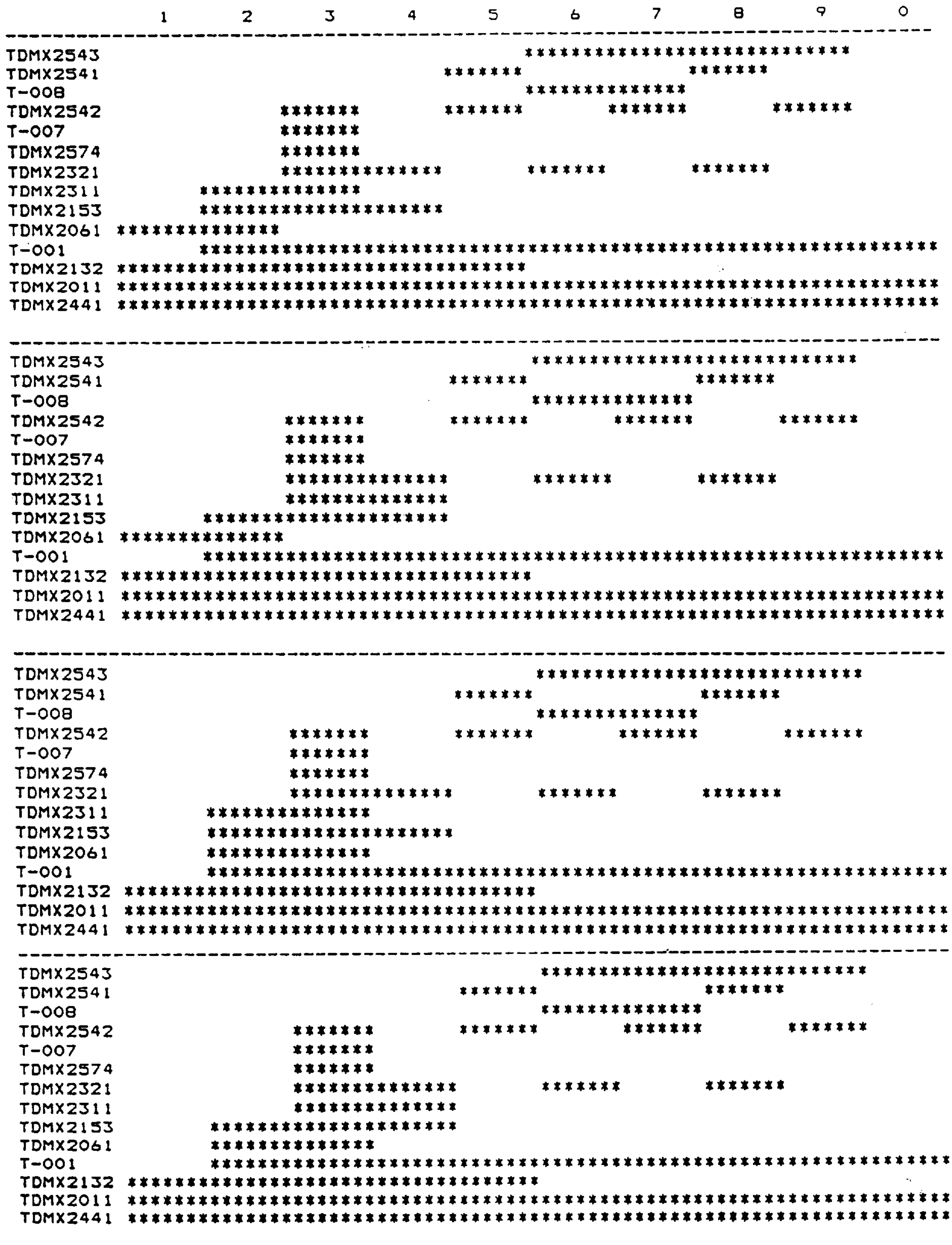


FIGURE 4 - AVERAGE TOTAL RESOURCE CONSUMPTION AND LIMITS