EXPERIENCE WITH THE USE OF
SUPERCOMPUTERS TO PROCESS LANDSAT DATA

MARTIN OZGA

Research Section
Remote Sensing Branch
Statistical Research Division
Statistical Reporting Service
United States Department of Agriculture
Washington, D.C. 20250

March 1984

EXPERIENCE WITH THE USE OF SUPERCOMPUTERS TO PROCESS LANDSAT DATA

MARTIN OZGA

STATISTICAL REPORTING SERVICE, U.S. DEPARTMENT OF AGRICULTURE

ABSTRACT

The Statistical Reporting Service of the United States Department of Agriculture has had to process large amounts of Landsat data in its project to improve the precision of crop acreage estimates by using Landsat data. Supercomputers have been found to be very useful and cost-effective tools in handling this processing load.

I. INTRODUCTION

The Statistical Reporting Service (SRS) of the United States Department of Agriculture (USDA) uses Landsat data to improve the precision of SRS crop acreage estimates. This use of Landsat data supplements ground data obtained by SRS enumerators who visit farms in selected areas known as segments. In the SRS Landsat procedures, the ground data are used to label Landsat data for supervised classifier training via within-class-type clustering and are also used in the calculation of the Landsat-based estimates of crop acreage (Ozga, 1976). Where sufficient ground and Landsat data are available, all Landsat scenes for entire states are completely classified-- some scenes more than once in attempts to optimize the estimate. The most desirable Landsat scenes are those acquired late in the growing season. With delays in receiving these scenes and with a mid-December deadline to generate estimates, a large number of scenes must be processed in a relatively short time. In 1983, crop estimates were calculated for seven states-- Arkansas, Illinois, Iowa, Oklahoma, Kansas, Missouri, and Colorado (eastern part only)--for a total of 60 scenes. Each of these scenes was classified at least once and many more than once. Moreover, a large amount of clustering was required. In addition, a land cover study of Missouri was performed using 11 scenes of which 9 were multitemporal.

For this heavy, concentrated processing and data transfer load, SRS has found supercomputers to be very useful and cost-effective. For the purposes of this paper, a supercomputer is a Single Instruction Multiple Data (SIMD) machine characterized by very fast processing in a vector or parallel mode in which several items of data are being operated on simultaneously. The items of data in Landsat processing are, of course, the pixels. Due to the large amount of data handled, high I/O transfer rates are necessary for Landsat applications.

Present supercomputers achieve parallel processing either by pipelined operation or by use of multiple processors. The two series of supercomputers currently commercially available, the CRAY series from CRAY Research, Inc., and the CYBER 200 series from Control Data Corporation (CDC), are pipelined machines. Examples of multi-processor machines include the ILLIAC-IV (now out of existence) and the experimental Massively Parallel Processor.

Typically, supercomputers have a 64-bit word size to facilitate scientific computation. Some have a half-word or 32-bit mode which runs faster for programs not requiring such high precision. Many Landsat processing programs are among that group.

A supercomputer will often have a front end processor. The front end processor, typically a mainframe or a super-mini, is used to gather data from various inputs (such as magnetic tape) and transfer it to the high speed disk connected to the supercomputer. Output data is collected from the supercomputer disk and either displayed for the user or saved for further processing. The high cost of supercomputers often makes it too costly for the supercomputer to do such data handling directly.

USDA-SRS used the ILLIAC-IV until it was discontinued and has since switched to its replacement, a CRAY X-MP. In addition, SRS has investigated the CYBER 200 series and plans to investigate the Massively Parallel Processor.

II. THE ILLIAC-IV

The current SRS remote sensing project has evolved, in part, from a project at the University of Illinois to investigate the use of the ILLIAC-IV for Landsat data processing. The ILLIAC-IV was developed by the University of Illinois under contract with the Department of Defense and was actually constructed by Burroughs Corporation (Hord, 1982). The ILLIAC-IV consisted of 64 processors each of which could access directly 2048 words in a shared memory. It had the usual 64-bit word, but could operate in 32-bit mode giving nearly the effect of 128 processors. This configuration was excellent for a maximum likelihood classification algorithm since the pixels are treated independently. However, for efficient operation, the mean and covariance matrices for all categories had to be stored in the memory of each processor thus limiting the number of categories although the limit chosen (64) was generally sufficient. The ILLIAC-IV had a high speeed disk, adequate for handling one full frame of MSS data.

The ILLIAC-IV was installed at NASA-Ames and became quasi-operational in the mid-to late-1970's. It was plagued by hardware problems due to components which were then obsolete. Since only one ILLIAC-IV was ever built, maintenance costs were high. The front end processor, a DEC-10 (TENEX) was inadequately configurated for Landsat processing since it provided neither the large storage nor the high speed transfer rates necessary for the large amounts of data required. The front-end DEC-10 was also heavily used for other processing. This situation improved when direct transfer from tape to ILLIAC disk was implemented. However, neither the disk nor the main memory was big enough for more than one job at a time so that users were charged full processing rates for time spent in data transfer. In spite of all these problems, SRS was able to do a significant amount of processing on the ILLIAC-IV and it provided better facilities than other machines available at the time. Typically, 5 to 10 minutes were required to classify a scene, depending on the number of categories. This included data transfers to and from ILLIAC disk but not data transfer from tape.

The ILLIAC-IV was discontinued in September, 1981, and replaced by a CRAY 1-S since upgraded to a CRAY X-MP.

III. CURRENT OPERATIONS ON THE CRAY

A. OVERVIEW

The CRAY is a pipelined machine built by CRAY Research, Inc (CRAY-1 S, 1981). It has the usual 64-bit word but does not support a 32-bit word mode. Further speedups are obtained by parallel operation of separate functional units and by "chaining" of vector operations. Vector operations are performed from eight 64-word vector registers. The main language used is FORTRAN-77. There are no syntactic extensions for vector operations. The compiler analyzes DO-loops to determine if they may be vectorized. In the case of nested loops, an attempt will be made to vectorize the inner loop only so loops must be coded carefully and in the proper order to be vectorized (CRAY CFT, 1981) (Petersen, 1983). Special vector procedures are available for some operations which may not be directly expressed in FORTRAN. Assembly language is available for writing functions which cannot be efficiently expressed in FORTRAN.

The CRAY at NASA-Ames currently has 2,000,000 64-bit words of main memory. Although not all of this is available for user programs (some being reserved for the operating system), memory size has not been a problem in any of our programs. There are also 1,000,000,000 64-bit words of high-speed disk storage. Most of this is available for temporary use by user programs although a small portion is taken up by permanent storage of user and system files. A MSS scene takes up about 5,000,000 words of this disk storage. The CRAY has several front end processors, a Control Data CYBER 170 and one or more VAX 11/780s. SRS uses a DEC-10 at Bolt Beranek and Newman (BBN) for most of its processing. This DEC-10 is connected to the ARPANET, a Department of Defense computer network, as is one of the VAXes at NASA-Ames. Smaller files are transferred directly to the VAX disk to be transferred to the CRAY when called for in jobs. Larger files, that is full scenes, are mailed on tapes. These are stored at NASA-Ames and, when called for in CRAY jobs, are read by the CYBER with the contents sent to the CRAY disk.

Currently, SRS uses the CRAY for maximum likelihood classification, the CLASSY clustering algorithm, aggregation of classified files, and block correlation to create multitemporal scenes. The classification program is available in two versions: a four-channel version which is heavily optimized and a two to sixteen channel version which is somewhat less optimized but still quite efficient. The aggregation program is not a vectorized program but rather is kept on the CRAY since it takes the large classified file along with other inputs and creates a small tabular file. The CLASSY clustering program has so far not been effectively vectorized, but since the CRAY is the only really powerful machine to which SRS has access, it is more efficient to use CLASSY on the CRAY than at BBN. The block correlation program has been very efficiently implemented on the CRAY since the correlation process lends itself well to vectorization.

B. MAXIMUM LIKELIHOOD CLASSIFICATION

The maximum likelihood classify program is a nearly ideal program for any sort of vector machine. The algorithm is simple and it is performed independently on a large number of identical format data elements (the pixels).

Basically, the process applies a discriminant function for each class (category) and assigns the pixel to the class yielding the highest value of the discriminant function. The discriminant function is:

$$G(X, I) = B[I] - .5[(X - M[I])^T V[I] (X - M[I])]$$

where X is the pixel,
I is the category
M[I] is the vector of mean values
V[I] is the inverted variance - covariance matrix
B[I] = -.5 log (determinant (V[I]))

In analyzing this problem for vector application, one may consider vectorizing over the categories or vectorizing over the pixels. That is, the discriminant functions for all categories may be computed for a single pixel simultaneously or the discriminant function for a single category may be computed for all pixels simultaneously. The latter method works by far the best. The reason is that on a vector machine, best operation is achieved at full capacity operation in the vector units. On the CRAY, the vector length is 64, the length of the vector registers. Since the number of pixels, even in a single buffer, is much larger than the number of categories, there will be many iterations over the pixels in which 64 pixels will be processed. With categories, if there are fewer than 64 categories, there will be no iterations in which the full vector length is achieved. Otherwise, there may be half the iterations at the full vector length and the other half at very much less than the vector length. Of course, at exactly 64 categories an efficient implementation is obtained, but in practice this occurs infrequently as users select various numbers of categories to reflect what is of interest in the scene. Also, on serial machines, various optimizations of the classification algorithm have been implemented. Many of these are invalid on a vector machine and would actually slow it down. For instance, any sort of table lookup scheme to avoid re-classifying the same pixel is not valid on a vector machine since in any collection of pixels some would be classified and others not. It is actually more time consuming to do all this checking and shuffling of pixels than to do the actual classification, particularly since machines such as the CRAY have very fast floating point operation. In the following description, we will consider only the four channel classify. As an exercise to test the capabilities of the CRAY, the main loops in this program were written in assembly language, although this yielded only a 20% improvement over the equivalent FORTRAN code.

The first step is the inversion of the covariance matrices for the categories and computation of determinants. This is a simple computation only done once so no attempt was made to vectorize it.

The Landsat data tapes are read in two possible formats. In the first, the tape is read in blocks of 512 words, that is 1024 four channel pixels per block. In the second, each record contains one line of the data. The Landsat data in our application are in a band interleaved by pixel format in which all four channels associated with a single pixel are in contiguous bytes. For further computation, it is necessary that each channel be extracted from its one byte position, and converted to floating point in a full word. There is no good way to do this in FORTRAN, so it was done in assembly language. It is convenient to have the four channel values of a pixel in contiguous words. The CRAY has a very nice feature, not often available on vector processors, which allows transfer between vector registers and groups of words with any constant increment in addresses. If this constant increment is one, the words are contiguous. A vector register is loaded with 64 words which is 128 pixels or 512 channel values. For each of the eight bytes in a word of a vector register the byte is positioned in the eight least significant bits of the same word in another vector register and masked so the other 58 bits are set to zero, converted to floating point, and then stored in the appropriate output words of memory with an increment of eight.

The next step is the computation of the discriminant function for each class for all pixels in the buffer. This may be easily coded in FORTRAN provided one is careful to make the inner loop be the loop on pixels. On a serial machine one would likely have the inner loop on channels which would not vectorize properly on the CRAY. A little analysis based on expanding the discriminant function formula reveals that fourteen floating point multiplications and a similar number of floating point additions or subtractions are required. Since the CRAY has eight vector registers, it is possible to store the data for each channel in a vector register for the duration of the computation and bring in elements of the covariance matrix into other vector registers as needed. Taking advantage of addition/subtraction and multiplication being in different functional units and of chaining leads to a very efficient implementation.

Once the value of the discriminant function has been computed, it must be compared with the previous value for that pixel so that the highest value and corresponding class are saved. It would be easy, but not efficient, to do this in serial mode. In CRAY FORTRAN there is a built-in function to do a comparison and selective store. However, the comparison has to be done twice since only one value may be stored for each call and we must store two values, the class and the value of the discriminant function. In assembly language, the mask register is loaded based on the comparison and values are stored into memory only from elements of the vector register for which a bit is set in the mask register.

The final step is the repacking of the data. Each class is stored in one byte. This is a limit of 255 classes, far more than are used in practice. The packing procedure is not readily expressed in

FORTRAN, so it is done in assembly language. The contents of a vector register are built from 512 words by taking the classes from words at an increment of eight and shifting them into the appropriate positions.

Some typical CPU times and costs for full scene classifications of MSS data are 109 seconds for 35 categories, 64 seconds for 13 categories costing $32.00, and 55 seconds for 11 categories costing $27.50. For multitemporal (8-channel) data a sample timing is 763 seconds for 89 categories and 5.37 million pixels costing $345.94. Classification times are proportional to the number of categories with some overhead, such as data unpacking and repacking, always included. For block correlation (correlation on a single channel of 32 by 32 blocks in 64 by 64 blocks), a sample timing is 11 seconds for 340 block pairs at a cost of $8.96 (including data pre-processing). From these timings, it is evident that what seemed to us a large processing load is actually quite small in comparison to the capacity of the CRAY. Even with a large number of CLASSY cluster jobs, we used only a small portion of that capacity and usually enjoyed quite rapid turnaround when the CRAY was operating properly, the usual case. Although costs vary from facility to facility, we found the CRAY to be cost effective and believe it would continue to be so in a commercial environment even with perhaps somewhat higher costs.

IV. CYBER 200

As part of an investigation on the use of other supercomputers, USDA-SRS contracted with CDC to test the classification program on the other commercially available supercomputer, the CYBER 200 (since succeeded by the similar CYBER 205) (Chase, 1982). Although, also a pipeline type machine, the CYBER 200 is somewhat different from the CRAY. Rather than operating from vector registers, the vector or pipeline unit(s) take their operands directly from memory using descriptors containing the starting address and number of words. This allows vectors to be very long, unlike the 64-word limit imposed by the vector registers on the CRAY. However, the vectors must be contiguous unlike the CRAY. It appears that the CYBER 200 can operate faster than the CRAY on very long vectors (certainly the case in the Landsat classify program) but is slower in scalar operations and short vectors (the former, at least, is also a part of the classification program). The CYBER 200 has the usual 64-bit word, but does support a 32-bit mode.

The CYBER 200 is programmed in an extended FORTRAN. Parallel arithmetic operations are expressed as arithmetic operations on descriptors. Descriptors are expressed as a symbolic address (i.e. the name of a FORTRAN variable) and a number of words. Other parallel

operations, such as tests, are handled by calls to a fairly large library of special procedures having distinctive names.

The CYBER 200 does not seem to have a front end processor but rather appears to be self contained, at least at the CYBERNET facility used for the test. In practice, it would be connected to a network to allow users to do serial processing and store files on other machines.

It is a little difficult to compare timings on the CYBER 200 since the CDC analysts combined the classify and aggregate programs into a single program and only classified that portion of the scene overlaid by the aggregation masks whereas on the CRAY the entire scene is classified. However, since the aggregations are typically short on the CRAY (one or two seconds each), most of the time is undoubtedly spent in classification. In scenes in which the aggregation masks covered most of the scenes, the CDC analysts reported job times, including aggregation, of 61.4 seconds for 12 categories and 78.6 seconds for 34 categories. For 8-channel (multitemporal) data, CDC reported 205.1 seconds for 30 categories and 256.6 seconds for 38 categories. These times appear to be comparable with those achieved on the CRAY and perhaps a little better in the 8-channel case (the 8-channel CRAY classification program could probably be speeded up with a little more programming effort).

V. MASSIVELY PARALLEL PROCESSOR

The Massively Parallel Processor (MPP) is an experimental supercomputer developed by Goodyear Aerospace Corporation for NASA-Goddard (Reeves, 1982). It consists of 16,384 processors arranged in a 128 by 128 grid. Each processor is quite "small" in that the individual operations are on operands of a single bit rather than on bytes or words as on conventional machines. Each processor has only 1024 bits of local memory. In addition, there is a two-megabyte high speed staging memory which the processors access frequently for data. The current front end is a VAX and data must be transferred between the VAX disk (or perhaps tape) through low speed channels to and from the staging memory. For Landsat processing, a great deal of movement must take place since a MSS scene requires about 40 megabytes.

The MPP is programmed in an extended PASCAL which has special syntax for expressing parallel operations. This is perhaps a better solution than the use of special procedure calls as on the CRAY and the CYBER 200.

At the time of this writing, it has not been possible to do very much testing on the MPP for Landsat processing. Very preliminary tests would indicate perhaps a 3 to 5 speedup over the CRAY for classification. Greater speedups could probably be attained by abandoning the use of 32-bit floating point in the classify algorithm, which does not

require high precision, and replacing it with strictly integer computation, but such experiments will have to be tried in the future. In its current configuration, the MPP is not adequate for the type of processing done by SRS due to lack of data handling capability. Such enhancements would not require any new technology, but only the type of storage and channel facilities generally available at CRAY and CYBER 205 sites.

VI. CONCLUSIONS

Supercomputers have been very useful to SRS for Landsat processing. SRS's processing load which once seemed heavy now seems rather light in comparison to the capabilities of modern supercomputers. SRS is just starting to work with TM data, but anticipates no problems in handling TM data easily on the CRAY. However, in order for a supercomputer to be effective for Landsat processing, adequate data handling and storage facilities, as well as high speed operation, are required.

Since it is not possible to justify the cost of a supercomputer solely for SRS processing, SRS has had to purchase time on those owned by someone else. Currently, this is at NASA-Ames. However, it seems that any facility providing a CRAY or CYBER 205 with suitable data handling facilities and network connections to other machines for serial processing would be quite adequate.

Finally, an often heard argument against use of supercomputers is made by those owning large minicomputers (such as VAXes) or even the new 32-bit "super-micros". The argument is that since they own the computer already, why not just start up a full-scene classification and let it run all night. This is fine if the total processing is just a few scenes and well spread out in time. However, SRS has many scenes concentrated late in the year. A failure on some night (due to hardware error, power failure, etc.) could cause serious problems in meeting deadlines. Also, the SRS estimation process requires a great deal of serial computation much of which can be done in batch mode and would thus compete with classification (and clustering). Finally, the relatively slow data rates of these machines would cause problems, especially with TM data.

ACKNOWLEDGEMENT

The author would like to thank Robert Slye and Don Card of NASA-Ames for their work on various of the CRAY codes mentioned in this paper.

REFERENCES

Chase, Sally. "CYBER 200 Bulk Landsat Processing Study, Final Report", Control Data Corporation, Rockville, Maryland, 1982.

CRAY-1 S Series Hardware Reference Manual, Cray Research, Inc., Mendota Heights, Minnesota, 1981

CRAY FORTRAN (CFT) Reference Manual, Cray Research, Inc., Mendota Heights, Minnesota, 1981

Hord, R. Michael. The ILLIAC-IV, The First Supercomputer, Computer Science Press, Rockville, Maryland, 1982

Ozga, Martin, Walter E. Donovan, and Chapman P. Gleason. "An Interactive System for Agricultural Acreage Estimates Using Landsat Data", Symposium Proceedings,Machine Processing of Remotely Sensed Data, Laboratory for Applications of Remote Sensing, Purdue University, West Lafayette, Indiana, 1977, pp. 113-123.

Petersen, W.P. "Vector FORTRAN for Numerical Problems on CRAY-1", Communications of the ACM, vol. 26, no. 11, November, 1983, pp. 1008-1021.

Reeves, Anthony and John D. Bruner. "The Language Parallel PASCAL and Other Aspects of the Massively Parallel Processor", School of Electrical Engineering, Cornell University, Phillips Hall, Ithaca, New York, 1982.

Martin Ozga has a B.S. in mathematics from the University of Michigan (1965) and a M.S. in computer science from the University of Illinois (1970). He has been with the Statistical Reporting Service of the United States Department of Agriculture as a Computer Systems Analyst since 1978.