SOME IMPROVEMENTS IN THE TECHNOLOGY OF STRING MERGING AND INTERNAL SORTING

Martin A. Goetz
Applied Data Research, Inc.
Princeton, N. J.

GENERAL

Sort/merge programs for magnetic tape computer systems are of two basic classes: ¹

1. Digital (or Radix) and Collation

The digital sort is useful in only a limited number of cases and is not examined in this paper.

The collation sort is the more general type of sort and is composed of two basic sub-programs:

1. The first sub-program internally sorts a group of data. The group of data after being sorted is referred to as a “string” or “initial string.” Such a sub-program is referred to as an internal sort.

2. The second sub-program merges two or more strings. It will produce as output longer strings and will eventually produce one string which contains all the data. Such a sub-program is referred to as a “merge,” or “string merge.”

The input data enters the internal sort sub-program only once and the merge sub-program one or more times. The various sorting systems in use today all try to minimize the execution time of the merge sub-program. This is accomplished by maximizing the way of the merge while at the same time keeping input strings the same size thus resulting in an “effective” power of the merge equal to the way of the merge. ² The read-forward Oscillating Technique described in this paper is directed toward this goal.

The Von-Neuman (T/2),¹ Polyphase,²,³ and Cascade⁴ merge techniques begin after all the data is processed by the internal sort sub-program and the initial strings are placed on tape (Figure 1). The Von-Neuman Technique places the initial strings on only half the available tapes; the Polyphase and Cascade place data on all the unused tapes. Depending on the number of tape units, one of the merging techniques will prove superior over the others. The read-backward Oscillating Technique⁵ writes several initial strings, transfers to the merge sub-program, merges the strings and returns to the internal sort—hence the name Oscillating (Figure 2). As the number of tape units used for sorting increase, the effectiveness of the Oscillating Sort increases. Given T input tapes, the effective power of this technique is T-2. This technique, previously considered only for read-backward tape systems, is developed for read-forward only tape systems as described in the following pages.

¹ For a distinction between “way of the merge” and “effective power of the merge” please see glossary of sort/merge terms.
READ-FORWARD OSCILLATING MERGE

The read-forward Oscillating Merge allows the use of a preceding internal sort technique which produces variable size string lengths (e.g., Replacement-Selection\(^6\)). It also allows, of course, techniques which produce fixed-size strings (e.g., successive merging\(^7\)).

This paper presents an example in which the internal sort is a Replacement-Selection. The Replacement-Selection Technique used as an internal sort produces an initial string length almost twice the size of the memory available for sorting. Since the Oscillating Merge Technique must merge initial strings formed by this technique, it is important that we review the nature of the output produced by the Replacement-Selection Technique.

Given a memory for building strings that can hold \(R\) records, the expected string length for random data for the first string\(^7\) is 1.73\(R\); for successive strings, 2.0\(R\); without replacement. 1.00\(R\). Given \(T\) tape units, the string on the first tape is estimated to be 1.73\(R\); for the last tape unit, 1.00\(R\), for all other tape units, 2.00\(R\). This variability in string length reduces the advantages of the read-backward Oscillating Technique.\(^8\) As will be demonstrated, this variability also reduces the effectiveness of the read-forward Oscillating Technique.

The read-forward Oscillating Technique can also be used with an internal sort that produces initial strings of a fixed size. In this case, the initial string length is 1.0\(R\) for all tape units.

THE TECHNIQUE

It is more convenient to describe the technique when the initial string lengths produced by the internal sort are fixed in length (1.0\(R\)). It will then be shown how the read-forward Oscillating Merge operates when used with the Replacement-Selection Technique in which variable length strings are produced.

Given \(N\) tape units (\(T_1\) to \(T_N\)) available as work tapes. (In the example shown in Figure 3, \(N\) is equal to 5.)
1. Initial strings are written on all tape units (T₁ to Tₙ). After each string is written on successive tape units, the tape is rewound. The last tape unit is not rewound (Figure 3A).

2. Tape units T₁ to Tₙ₋₁ are merged onto tape units Tₙ, at which time all tape units are rewound (Figure 3B).

3. As soon as tape unit T₁ is rewound (this tape unit contains only one string of data), a string is written on tape unit T₁ and then it is rewound. Strings are consecutively written on tape units T₂ to Tₙ₋₁. All tape units except Tₙ₋₁ are immediately rewound (Figure 3C). Tₙ₋₁ is not rewound.

4. A merge of T₁ to Tₙ₋₂ and Tₙ is performed onto Tₙ₋₁. Then all tapes are rewound (Figure 3D).

5. The cyclings shown above continue for N cycles at which time all tape units contain a string length of size (N−1) times the initial string length (Figure 3E).

6. The (N−1) size strings are merged, the power being (N−1) as shown in Figure 3F.

7. The pattern shown continues until the data is exhausted, at which time one string will be formed, similar to the read-backward Oscillating Technique.

Note that after 16 strings were merged, the rewind interlock time was equal to rewind time for N−1 strings. This time is a relatively small portion of the rewind time. Note that rewind time is minimized by starting the development of the next string in a group, although it will not be merged until the next cycle (String 21 and Strings 17−20).

When using the Replacement-Selection Technique in conjunction with the read-forward Oscillating Technique, initial strings are constrained not to exceed twice the number of records (R) in memory. After 2.0R records are written on tape, the string is completed. Note that with other merge techniques the minimum string length might be 1.0R as a lower limit and “all” records to be sorted (an entire file) as the upper limit. When using the read-forward Oscillating Technique, the upper limit is set at 2.0R. When 2.0R records can not be formed, dummy records (and blocks)
are substituted to form 2.0R records. Since 2.0R is the expected length, excessive dummy will not be formed. 2.0R may not be an optimum figure. It will, however, closely approximate the optimum as the number of tape units increase.

To the degree that the read-forward selection technique must generate dummy records to produce string lengths of 2.0R, it is less efficient than the read-backward technique. As stated above, however, this reduction in efficiency should be slight under most circumstances.

THE PROBLEM OF WRITING ON TAPE USING THE READ-FORWARD OSCILLATING MERGE

The read-forward Oscillating Merge herein proposed requires that data be written on the front of a tape without destroying information further down the tape which will be subsequently read. Depending on the computer system and the tape units, this may cause problems. For some computer systems, the read head may not be positioned properly due to start-stop time variations, automatic bypassing of unwritable tape (bad spots), effects of the erase head or differences in writing density. For systems with a tape rewrite feature, there would be no problem. For systems which allow tape erase, gaps on the tape can be program generated which will solve the problem. If neither of these features are available "hash" blocks can be inserted to protect information which must be subsequently read. The technique may not be applicable to older systems where there is no programmed error control.

Because of the wide variety of tape systems, this problem is not covered in more detail. It has been investigated, and it can be shown that the additional programming to cope with this problem is trivial.

CONCLUSION

The technique described offers the same potential as the read-backward Oscillating Technique, namely: as the number of tapes increase, this technique will perform the sorting task almost twice as efficiently as the N/2 (Von Neuman) and more efficiently than the Polyphase or Cascade Merge Technique. The Read-Forward Oscillating Technique might be considered even in systems which allow backward reading. This is particularly the case in systems where tape reverse interlock is high.

The foregoing presentation of the read-forward technique should not be construed as a recommendation of this technique to the exclusion of other methods. The selection of proper sorting techniques is a complex problem dealt with in detail in other papers 5, 8, 9, 10 and which is not completely formalized at this time.

VARIABLE LENGTH RECORD SORTING USING THE REPLACEMENT-SELECTION TECHNIQUE

INTRODUCTION

Another area where efficiencies in sorting can be attained, is in the internal sort. Since the number of merging passes is based on the number of internal strings, it is desirable to minimize the number of strings formed by the internal sort. The number of strings are minimized when the amount of data sorted at one time (the length of the string) is maximized.

The sorting techniques in use today limit the string length for variable size records to the number of records that can be stored in memory. This section describes a sorting technique that permits an initial string to be formed that contains approximately twice as much data as can fit into the working storage available in memory.7

GENERAL

The variable-record length internal sort segment uses a modified version of the Replacement-Selection Technique.6 This technique has previously been applied to the sorting of fixed-size records or to a variable-size record converted into a fixed format.

The proposed technique temporarily "disjoins" a variable-size record into one or more fixed-size "pieces" (referred to as segments) and at selection time combines the separate segments of the record. No expansion of the record occurs.
The selection of an "optimum" fixed-size segment storage area is either determined by the user (based on his knowledge of the data) or assigned by the computer program. The optimum size is one which will produce the longest string on tape without causing average size records to be broken up into a large number of segments. Long strings are produced by selecting a fixed storage segment size into which the records will fit without the need for excessive "fill" when the last "segment" of the record is moved into the fixed segment area. The selected segment size should be such that all keys for each record appear in the first disjointed segment when subdivided.

A brief review of the selection logic for conventional fixed size Replacement-Selection internal sorting follows:

1. Core memory is divided into 4 parts.
   (1) Instructions
   (2) Input Areas
   (3) Output Areas
   (4) A string building area (work storage area)

2. The string building area is further subdivided into pockets (or slots) equal to a fixed size plus about 12 characters to hold information used during the selection process. The selection process is similar to a tournament match in which the selected record at each level is referred to as a "winner." Assume as an example, the string building area can contain 12 records. The logic for selection is as follows:
   1. A string building area is filled with records (R) from the input area. (See Figure 4.)
   2. The first record in the string building area is compared against the second, the third against the 4th, etc. In this manner a set of first round winners is selected. The addresses of the winners are stored.
   3. In a similar way succeeding rounds of winners are selected until one final winner is selected. Addresses of the winners of each round are recorded. This process is called initialization of the tree and requires R-1 comparisons. (See Figure 5.)
   4. The final winner is moved to the output and its position in the string building area is replaced with a new record from the input area.
   5. If the new input record could be put out in sequence with respect to the last record that was put out, it can participate in the current tournament. Otherwise, it may not participate in the current tournament.
   6. Each subsequent record selected requires \( \log_2 R \) comparisons. This selection process is referred to as a "scan."
   7. When no records can participate, the current tournament is over and a new string must be formed.

USING THE REPLACEMENT-SELECTION TECHNIQUE FOR VARIABLE LENGTH RECORDS

A record in the input area is divided so that one or more segments of it may be moved into
slots in the string building area. The first such segment, containing the key, is called the header; all other segments are trailers. Each segment of a record, except the last, contains as part of its indicative information, the address of the segment of the record which is its immediate successor.

When the string building area is completely filled it is "initialized." Only headers participate in this initialization and only headers are winners in each "round" of the tournament. Initialization is concluded when a final winner has been determined.

At this point, the winner is moved to the output area. Immediately, its place in the string building area is filled from the input area. Whenever a "segment" is moved from the string-building area, that slot is immediately filled. This is true irrespective of whether a header or trailer segment was moved out (Figure 6). As segments are moved out they are rejoined in the output area into a variable length record. Any "fill" in the last segment is deleted at this time.

After a string building area slot has been refilled, a "scan" takes place if one is necessary. It is necessary to "scan" if either a header was moved out of the string building area or if a header was moved into the string building area. In other words, the only time a scan is not performed is when a trailer is replaced by a trailer.

After the scan, the next "segment" is moved into the output area according to the following rules:

1. If the last segment moved to the output contained a reference to a successor (which must be a trailer), the successor is moved. This accomplishes the result of assembling the records which were segmented initially.
2. If the last segment moved was without a successor, the current final winner of the tournament (which must be a header), is moved to the output.

CONCLUSION

The logic for processing variable-length records using the Replacement-Selection Technique requires an additional address in the tournament tree which is used to chain between segments of a variable-length record. Additional logic is required to process variable length records. Records are segmented into header and trailers. After the selection process, they are recombined into a variable length record. The logic of the Replacement-Selection Technique requires modifications as described. As in the Replacement-Selection Technique for fixed size records, data is moved only twice. The overall logic of the Replacement-Selection Technique is retained and all its advantages are exploited.

ACKNOWLEDGEMENT

The author wishes to express his appreciation to Mr. Warren F. Spalding of Applied Data Research, Inc. for his assistance in the organization and editing of this paper.

GLOSSARY OF SORTING AND MERGING TERMS USED WITHIN THIS PAPER*

Backward read
A feature available on some magnetic tape system whereby the magnetic tape units can transfer data to computer storage while mov-
ing in a reverse direction. Normally used, if available, during the external sort phases to reduce rewind time.

**Balanced sorting**
See: (T/2)-way merging.

**Cascade merging**
A technique used in a sort program to merge strings of sequenced data. Given T work tapes, merging is performed at T-1 on part of the data, T-2 on parts of the data, and so on. Strings of sequenced data are distributed in a Fibonacci Series on the work tapes preceding each merge. The effective power of the merge varies between T-1 and T-2 but in all cases is less than the power of the Polyphase Merge. Cf: effective power of the merge.

**Collating**
Sequencing a group of records by comparing the key of one record with another record until equality, greater than, or less than is determined.

**Collating sequence**
The sorting sequence; a description of the sort key for a file of records.

**Collating sorting**
A sort which uses a technique of continuous merging of data until one sequence is developed.

**Computer limited**
See: process limited.

**Digital sorting**
A sort which uses a technique similar to sorting on tabulation machines (e.g., IBM Sorter). The elapsed time is directly proportional to the number of characters in the sequencing key and the volume of data. Also "radix sort."

**Effective power of the merge**
Equal to S_S, where S is the number of input strings and N is the average number of times each element of data is read.

**Fibonacci series**
A series where the current number is equal to the sum of the two preceding numbers: i.e., 1, 2, 3, 5, 8, and so on. Some sort programs distribute strings of data onto work tapes so that the number of strings on successive tapes form a Fibonacci series.

**Fixed size records**
Denumerable file elements each of which has the same number of words, characters, bits, fields, etc. Cf: variable-length records.

**Generalized sort**
A sort program which will accept the introduction of parameters at run time and which does not generate a program.

**Generated sort**
A production program which was produced by a sort generator.

**Input tape(s)**
Tape(s) containing a file in arbitrary sequence to be introduced into a sort/merge program.

**Insertion method**
See: Sifting.

**Item**
See: Record.

**Key**
Also, sequencing key; criteria; sequencing criteria. The fields in a record which determine, or are used as a basis for determining, the sequence of records in a file.

**Magnetic tape sorting**
A sort program that utilizes magnetic tapes for auxiliary storage during a sort.

**Major key**
The most significant key in a record.

**Merge**
A program that performs merging.

**Merging**
The forming of a single file of sequenced records from two or more files of sequenced records.

**Multi-file sorting**
The automatic sequencing of more than one file, based upon separate parameters for each file, without operator intervention.
**Multipass sort**
A sort program which is designed to sort more data than can be contained within the internal memory of a central computer. Intermediate storage, such as disc, tape, drum, etc. is required.

**Optimum merging patterns**
The determination of the sequence in which specific sorted tapes in a file should be processed so as to minimize the total number of merge passes required to create a single file of sequenced records.

**Order of the merge**
The number of input files to a merge program. Also: power of the merge.

**Oscillating merge**
A technique used in a sort program to merge strings of sequenced data. For tape systems that permit backward reading, the effective power of the merge is equal to T-2.

**Output tape(s)**
Tapes containing a file in specified sequence as a result of a specific sort/merge process.

**Pass**
The processing of each file record once for the purpose of reducing the number of strings of sequenced records and increasing the number of sequenced records per string.

**Phase**
An arbitrary segmentation of a sort program. Many sorts are segmented into: three phases: initialization phase, internal phase, merge phase.

**Polyphase merging**
A technique used in a sort program to merge strings of sequenced data. Given T work tapes, merging is performed at the power of T-1. The effective power of the merge varies between T-1 and T-2 depending on the amount of input data and the number of strings.

**Power of the merge**
Also: way of the merge; order of the merge, the number of inputs to a merge program. Cf: effective power of the merge.

**Process limited**
Also: computer limited. A sort program in which the execution time of the internal instructions determines the elapsed time required to sort. Cf: tape limited.

**Radix-sort**
See: Digital sorting.

**Record**
The basic element of a file such that the sorting of file constitutes the re-ordering of file records; also referred to as “item.”

**Replacement-selection technique**
A technique used in the internal portion of a sort program. The results of the comparisons between groups of records are stored for later use. A selected record is placed on the output tape and a new record replaces the selected record. Given N records, a record is selected with \(1 + \log_2 N\) tests; the expected string length for random data is \(2N\) records.

**Rewind time**
Elapsed time consumed by a sort/merge program for restoring intermediate and final tape files to original position.

**Scratch tape(s)**
See: Work tapes.

**Sequence break**
That point in a file between the end of one string and start of another.

**Sequencing criteria**
See: Key.

**Sequencing key**
See: Key.

**Sifting**
A method of internal sorting where records are moved to permit the insertion of records; also called “insertion method.”

**Sort**
The copying of a file of records into a corresponding file in a specified sequence.

**Sort, external**
The second phase of a multipass sort program, wherein strings of data are continually merged until one string of sequenced data is formed. Cf: string merge.
Sort, internal
The sequencing of two or more records within the central computer memory; the first phase of a multipass sort program.

Sort generator
A program which generates a sort program for production running.

String
A group of sequenced records, normally stored in auxiliary computer storage; i.e., disc, tape or drum.

String merge
Program that performs merging.

String merging
The forming of a single string from two or more strings of sequenced records.

(T/2)-way merging
A technique used in a sort program to merge strings of sequenced data. The power of the merge is equal to T/2.

Tape limited
Also: I/O limited. A sort program in which the effective transfer rate of tape units determines the elapsed time required to sort. Cf: process limited.

Tennis match sorting
See: Replacement-Selection Technique.

Tournament sorting
See: Replacement-Selection Technique.

Variable-length records
Denumerable file elements for which the number of words, characters, bits, fields, etc. is not constant. Cf: fixed-size records.

Von Neuman sort
See: (T/2)-way merging.

Way of the merge
See: power of the merge.

Work tape(s)
Also: scratch tapes. Tape(s) used to store intermediate pass data during a sort program.

Xmas tree sorting
See: Replacement-Selection Technique.

REFERENCES