Whatever Happened to CMAP for Macintosh?

A Status Report

Peter Castine
H5/70 GmbH
Tiernstraße 7-73
10551 Berlin, Germany
e-mail: post@mv2.kgw.to-berlin.de

Abstract

The Contemporary Music Analysis Package (CMAP) is a set of programs, originally implemented for the Unix operating system, to assist musicologists in analyzing and composing music. CMAP provides a comprehensive set of tools for analyzing and composing music, including operations on collections of pitch classes. Many characteristics of CMAP, however, are not fully implemented, and the user interface is not as advanced as in Unix.

1. Background

The Contemporary Music Analysis Package (CMAP) is a computer-based tool for the composition and analysis of contemporary music. CMAP provides functions to analyze and manipulate collections of pitch classes, analyze K/R relationships, and calculate invariances among collections of pitch classes. Functions are provided for all standard set and row operations.

The original version of CMAP was developed by Craig Harris and Alexander Brinkman for the Unix operating system and is described in detail in Harris' Ph.D. thesis (1986) and several articles written by Harris and Brinkman (1986).

Many characteristics of CMAP are not fully implemented, and the user interface is not as advanced as in Unix. The possibility of writing shell scripts makes the solution of problems not envisaged by the original program feasible. Using this feasibility to its fullest requires the user to expend considerable time and energy to become proficient with Unix. Many of the programs are set by assumption that an ANSI terminal is being used.

Figure 1: The Analysis Scratchpad. The first two rows were generated by a CMAP command that searched for all window contiguity fives that contain all six intervals. The union of these two sets was calculated for the third line. The following three lines show information calculated and entered into common blocks. The seventh line contains pitch classes entered by the user and a pc count generated by CMAP. The last line contains a field for data entry.

2P.08 369 ICMC Proceedings 1993
(for instance, pc matrices are effectively limited to fifteen-tone rows for on-screen display). Finally, most musicians want something they can use on their machine at home.  

2. CMAP for Macintosh

A version of CMAP is currently under development for the Apple Macintosh. One of the primary concerns during the design stage of the CMAP for Macintosh project was to incorporate the functions provided by CMAP in a unified, "Macintosh-like" user interface.

2.1 Functions

Most functions of CMAP on the Macintosh are available through the 'Analysis Scratchpad.' This is a window in which pitch class combinations can be entered, analyzed, and manipulated. Set information is displayed in analysis lines that show: pc's entered; set class name; prime form; interval class, inversion, and adjacent interval vectors. The basic form of the display is a tabular representation, with one pitch class combination per line and the set class information organized in columns. The user can generate sets according to various abstract criteria (for instance, all prime forms of sets containing each interval class exactly once). Further commands are provided for generating new sets based on sets already in the Scratchpad, using twelve-tone operations, set operations (union, intersection, etc.), partitioning and unification. All of these functions are accessed through direct manipulation and menus, dialogs are provided for specifying parameters.

However, many of CMAP's functions, such as those involving row matrices, set complex tables, subset relations, and pitch class mapping, provide information that does not fit into the Analysis Scratchpad.

Two methods have been found for dealing with these functions.

First, certain commands simply append a comment on to an analysis line. For instance, 'Count Pitch Classes' appends a table of the frequency with which each pitch class appears in a tone row; analysis lines generated with the 'Partition...' command include information regrouping the set from which the partition was generated. As an extension of this idea, the user is also able to append comments to lines in the Scratchpad. This can be an aid in referencing sets to a musical work being analyzed.

Certain commands provide more information than that which can be conveniently displayed as a comment to an analysis line. For these commands additional windows were designed. These include a 'Row Matrix' window, in which pitch class and pitch order matrices can be generated. This window is also used for calculating invariance matrices and working with rotational arrays. The user can search for the occurrence of specific sets contained in the matrix. Other windows have been designed to generate tables of relations between sets ('Set Relations Window'); and to show operations upon sets which a set maps into itself or into an arbitrary superset ('Set Mapping Window').

2.2 Data Entry

The Unix version of CMAP requires the user to enter pitch class data using hexadecimal notation. Although this is an extremely efficient way of entering data, many users prefer to view musical data in other

1 The original CMAP only provided a function to calculate set complex relations: CMAP/Mac has extended this with the capability of calculating Forte's R1, R2, R1, and R2 relations as defined in [Forc. 1973].
formats. The Macintosh version of CMAP supports hexadecimal notation, decimal integers, and pitch class names; MIDI keyboards are also supported for data entry. The notation used during data entry may be different from the display format.

2.3 Combining Commands

One of the challenges involved with porting a package of Unix programs to Macintosh, user interface considerations aside, is in providing a mechanism for the individual components of the package to communicate with each other.

When CMAP/Mac was first designed, System 7 was hardly more than, and Applescript was less than, a rumor. The generic commands Quit, Copy, and Paste were the primary means of data exchange. To make the use of these commands more efficient, CMAP/Mac was designed as a single application. Furthermore, the result of every command is always automatically selected, so that the data produced will be available for manipulation by the next command chosen. This allows sequences of commands to be automated by any of the journaling utilities provided for Macintosh, such as QuickKeys.

The version of CMAP/Mac currently under development supports only required Apple Events. Future versions will support core Apple Events and AppleScript, providing a rich syntax for automating tasks.

2.4 A Word on Implementation

All versions of CMAP are conceived as tools for interactive analysis, and this is particularly true of the Macintosh version. To provide a fast and efficient implementation of the various software tools, a database of set class information is stored in an extremely compact form which remains resident in memory at all times. This database of class model is called the Set Class Table. All sets are stored as bit vectors. This limits memory usage to a minimum and allows for the implementation of particularly efficient algorithms for all operations. In the Macintosh version, a further space optimization was implemented by building a compact index into the Set Class Table. The cost to meet additional memory requirements is extremely modest, and was balanced by a slightly more space efficient representation of the Set Class Table than that used in the Unix version.

3. Conclusions

The Macintosh version of CMAP provides a comprehensive set of tools for set-theoretic analysis, unifying a collection of almost thirty separate programs into one unified application. Priority has been given to implementing fast algorithms using compact data representations. A further goal was to provide the same flexibility found in the Unix version under an operation system that did not inherently support scripting.

References

[Fonte, 1978]
[Harris and Brinkman, 1986]
[Harris, 1986b]
[Harris and Brinkman, 1989]

Figure 4: Set Operations Menu, provides most of the functions used in the Analysis Scratchpad.

ICMC Proceedings 1993