Chapter 2

Introduction

2.1 Purposes of the research

The purposes of this research lie in the following aspects:

➢ For music analysis
  First of all, the entire original music can be understood more sharply and deeply by decomposing the whole image of music into individual components and reexamining the structures and the forms which compose it. Music analysis is indeed important for correctly understanding music and the performance [Coo60]. It plays one of the most important roles in a musical information processing system [Kel87][Pal89].

➢ For generating expressive musical renditions
  Music is used to express our feelings. Same as language, music also has expressions. For the performance of music, the height of the pitch, the urgency of the speed, the loudness of the volume express different meanings. Unexpressive music, like unexpressive language, is dry as dust. The existing automatic performance or accompaniment system chiefly obtains performance information from score data, then gets mixed up with the player's performance information and the system designer's experience. Since different melodies have different characteristics, different players use different performance expressions, and different system designers have different experiences, the system's expressions disperse again and again in different phases. Thus it's difficult to apply abundant expressions to musical renditions [Wim93] [Wid94].

➢ For acquisition and use of the performance analysis and rules
  Like language, music has rules of expressions. In project PSYCHE, with many years of research, a large amount of performances by various musicians have been analyzed. And some performance rules concerning expressions have been found. If these rules are
applied to the performance, a better expression can be achieved. However, the existing independent systems in PSYCHE use their own methods to analyze rules. To use each other's results is also difficult.

➢ For performance evaluation

Formerly, the results of an automatic performance are being evaluated by users through listening. Like the process of analyzing musicians' performances, if the performance data can be visualized on the screen and compared with musicians' performances, the defects will be easier to find.

➢ Necessity of an integrated system

Finally, in project PSYCHE, various computer systems have been constructed up to now. Among them, there are automatic performance generation system, automatic accompaniment system, and human-performance analysis system, etc. These systems exist independently, can not use each other's results and common information. The integration of the whole project is poor. In order to let these existing systems and potentially new sub-systems to use each other's results and information mutually, it is necessary to develop an integrated system.

Due to the above mentioned current states and purposes, we've developed the computer assisted music analysis system - DAPHNE. The music analysis information obtained by DAPHNE can be used by other existing systems in PSYCHE. As a result, by applying the rules (obtained by analyzing the performances of players) to music analysis information, expression information can be added to the score data.

2.2 Related researches on music analysis

Computer-assisted music analysis has been researched for almost 50 years. It has been developed under various premises with a variety of methodologies [Alp80]. As much as I know, there are only a few computer-assisted music analysis systems. These systems can be classified based upon three epistemological levels [Nic00] [Oka98].

① The first is the music genre, such as "classical" or "pop". It is the basis for analysis.
② The second is the form in which music is analyzed. It can be classified by "notation-based
analysis" and "performance-based analysis", and applies analysis methods strongly.

The third is computer-assisted analysis method. The following methods have been used up to now. Some of them can be divided into sub-methods.

- Statistical and Information-Theoretical Analysis
- Set Theoretical Analysis
- Other Mathematical Analysis
- Hierarchical Analysis
- Transformational Analysis
- Schenkerian Analysis
- Spectral Analysis
- Cognitive and AI Analysis
- Combined Analysis

Every method of music analysis has its advantages for certain goals of the analysis. On the other hand, every analysis method has its limits [Des94] [Hor93].

In this section, I will introduce some related music analysis systems and theories.

2.2.1 Humdrum

David Huron developed humdrum in 1993. It has been used by lots of researchers now [Hur]. This system is a powerful analysis system whose purpose is mainly score analysis that does not intend music creation. Thus it has a strong set of functions, specified by a line command style, for analyzing musical scores with statistics such as to finding the similar patterns in rhythm or melody (analyzing Leitmotiv of the whole piece of is an example of using Humdrum), but obtained information is not intended for musical performance rendering nor composition. Humdrum allows users to pose and answer questions such as the following:

- Identify the most common word following 'gloria' in Gregorian chants.
- Identify all works that end with a 'tierce de picardie'.
- Is German drinking songs more likely to be in triple meter?
- ....

There are currently roughly 10,000 encoded works consistent with the Humdrum syntax. This system verifies the extraction and the hypothesis of the rule by retrieving and comparing patterns of the tune by using the statistical method. Since Humdrum is a general-purpose system, it only uses a statistical method, does not touch about the music theory [Hur89]. A set of commands of Humdrum-format and a database of encoded piece is necessity when analyzing.

2.2.2 GTTM
The Generative Theory of Tonal Music (GTMM) was developed by Fred Lerdahl and Ray Jackendoff as an approach to the analysis of tonal music based on foundations of human perception [Ler83]. Combining the formal methodology and psychological concerns of Chomskian linguistics with the insights of Schenkerian music theory, GTMM attempts to describe how a listener experienced in the tonal idiom intuitively creates an understanding of a complete musical structure [Pee84]. The theory is composed of four main modules, each of which assigns a separate part of the structural description of a listener's understanding of a piece of music. Analysis. There is the metric description of the piece, the grouping description, the time span reduction and the prolongational reduction. Each component is first exposed informally and then expressed as a logically rigid set of "rules" of "musical grammar". The latter consist of both well-formedness rules, which describe the minimal conditions for an intuitively understandable structure, and preference rules, which correspond to the intuitions that allow a listener to choose the preferred interpretation of the structure from all of the possible ones that conform to the well-formedness rules [Tak98]. Here is an analysis rule of grouping structure,

Example: Consider a sequence of four notes $n1n2n3n4$. All else being equal, the transition $n2-n3$ may be as a group boundary if

a. (Register) the transition $n2-n3$ is greater intervallic distance than both $n1-n2$ and $n3-n4$, or if
b. (Dynamics) the transition $n2-n3$ involves a change in dynamics and $n1-n2$ and $n3-n4$ do not, or if
c. (Articulation) the transition $n2-n3$ involves a change in Articulation and $n1-n2$ and $n3-n4$ do not, or if
d. (Length) $n2$ and $n3$ are of different lengths and both pairs $n1, n2$ and $n3, n4$ do not differ in length.

Music is hierarchically analyzed from the note to a phrase by these rules. The four components are not independent as there are rules, which assign structure in one module by reference to structure in another. Because of the difficulty of rule's conflict and selection, a complete automatic analysis on the computer cannot be achieved yet, though there is universality in the application research on GTMM up to now.
2.2.3 Narmour

Eugene Narmour formulates a comprehensive theory of melodic syntax to explain cognitive relations between melodic tones at their most basic level. Narmour develops one scaled set of rules modeling implication and realization in all the primary parameters of music. Through an elaborate and original analytic symbology, he shows that a kind of "genetic code" governs the perception and cognition of melody. One is an automatic, "brute" system operating on stylistic primitives from the bottom up. The other constitutes a learned system of schemata impinging on style structures from the top down [Coo87] [Mur98] [Cun92].

The theoretical constants Narmour uses are context-free and, therefore, applicable to all styles of melody. He places considerable emphasis on the listener's cognitive performance (that is, fundamental melodic perception as opposed to acquired musical competence). He concentrates almost exclusively on low-level, note-to-note relations, which he typologies according to the degree to which these relations are similar or different, and hence anticipatable or unexpected.

Very approximately, relations within pairs of consecutive intervals can be classified as exemplifying either Process (if they are similar) or Reversal (if they are different), with small intervals giving rise to implications of continuation or process, and large intervals giving rise to implications of reversal, or differentiation. Similarity or difference can be found both in the domain of interval size and in the domain of registral direction (ascent versus descent). These basic terms provide means of classifying groups of consecutive notes, so an ascending passage such as c-d-e-f would constitute a Process (symbolized as P), an ascending-descending fragment going from a large interval to a small one such as c-g-f would constitute a Reversal (R), a figure such as c-e-c would constitute an Intervallic Duplication (ID), an ascending-descending fragment such as c-d-b would constitute an Intervallic Process (IP), while an ascending-descending figure such as b-f#-g would constitute a Registral Reversal (VR) etc.

The result is a highly generalized theory useful in researching all manner of psychological and music-theoretic problems concerned with the analysis and cognition of melody.

All of the above systems, same as DAPHNE, are based on notation. However, at present, there are only a few such kinds of systems. Most of other analysis systems are based on performance data. One example is the following system - POCO.
2.2.4 POCO

POCO is being developed by the "Music, Mind, Machine" research group at Nijmegen Institute for Cognition and Information. Different from the above systems, it is an evolving environment for analyzing musical performances, especially with regard to expressive timing. Some part of POCO is used for analyzing musical performance of MIDI data. It has support for program development and system maintenance, and enables the programmer who extends the system to ignore details about the multi-modal user interface and the documentation, since these are generated automatically [Hon90].

2.3 The originality of the research

Compared with other existing analysis systems, DAPHNE has the following originality.

First of all, based on the general music theory, DAPHNE's analysis results are similar to musicians’ analysis results. Therefore, to achieve a human-like expressive beautiful performance becomes possible.

Moreover, DAPHNE is not merely an analysis system. It is an integrated musical information processing system on both analysis and performance. The analyzed results can be automatically added to the performance data and directly used by the automatic performance generation system.

The performance analysis and performance information can also be obtained. The generated performance can be evaluated. Analysis information and performance information can be shared. The system’s functions are enhanced greatly for both music analysis and performance generation.

In addition, a flexible WYSIWYG user interface is adopted. Users do not have the sense of incompatibility in the analysis on actual score images because neither a special language nor complicated instructions is used. We expect DAPHNE to be used not only by computer scientists interested in music but also by music professionals.

The originality of DAPHNE will bring analysis systems to a new stage. And it has great meanings to the research intended for music.
2.4 Difficulties of the research

In this research, selecting the analysis theory, analysis algorithms design, and system implementation are all covered. There are some difficulties according to the goals and characteristics of DAPHNE.

2.4.1 Theories

On the theory aspect, the following two points are challenging:

- Selection of analysis theory
- Generation of analysis algorithms

For a musical information processing system, the underground music theory is necessary for generating algorithms. I designed most of the analysis algorithms in DAPHNE based on the music theories referring to preceding researches and opinions from musicians. Designing the algorithms is not easy because of the complexity and the vagueness nature of music analysis. For instance, in case of structure analysis, one of the basic rules is to compose a motif with two measures. But in fact, it is necessary to consider various detailed conditions like the influence of Aufakt, melody’s transformation from right hand to left hand, and the existence of stops. In case of musical form analysis, since the constructions up to sentence in the Phrasing theory are incomplete, it's necessary to create new structure units larger than sentence.

2.4.2 Systems

2.4.2.1 System design and implementation

As an integrated system to obtain musical information, it is necessary to take the following points into consideration:

- There are various data like scores and performances to be processed.
- The correspondence of various existing sub-systems in PSYCHE.
- Using different modules’ results mutually.

For instance, in case of analysis items, analysis results will be influenced by the system design since there are different levels of processing like management of internal data and algorithms, results exchange with other analysis items, and adding new analysis items, etc.

Moreover, because the range of obtained information is so wide, it is necessary to construct a relatively large system.

Besides, in this research field, there are almost no similar systems that can be compared or referenced.
2.4.2.2 User interface

Since data and diagram were the center of the former system’s user interface, it was criticized that the system was difficult to use, especially for non-computer experts. Based on this opinion, a score oriented user interface was developed. However, there are not many analysis systems using scores by far. One reason is the problem of score recognition. As one field of musical information processing, score recognition is still under researching and few practical products has been made [Mat98]. Therefore, it is necessary to find methods other than score recognition. The result is that the bitmap score images and the score description file are adopted.

2.5 The history of DAPHNE

DAPHNE has been developed for nearly five years. It is the compilation and enhancement of the research results of PSYCHE by far.

In 1996, Prof. Igarashi put forward the idea of DAPHNE: a system aims at music analysis. He chose the Phrasing theory as the theoretical base. Specifically, he proposed the interactive grammar analysis method with examples. In the meantime, Ms. Iyatomi posed the performance rule - "performances with similar structures are also similar", which made the relationship between music analysis and automatic performance become closer. She also did the pioneering development of DAPHNE using LISP under UNIX. At that time, I was in the first year of my master program and involved in verifying the performance rules by analyzing some performance samples.

In 1997, according to the characteristics of musical structures and the new progress of programming technologies, Ms. Hiraga brought object technique to the project. Under her directions, I designed the first specification of DAPHNE under Windows. Including the idea of analysis on the actual scores and the visualization of performance data. But this design was abandoned. One of the reasons is that it would cost too much time to construct such a system.

After that, Ms. Hiraga made the initial design of DAPHNE system. The system is composed of two parts: a Visual Basic (VB) coded user interface and a Visual C++ coded analysis kernel. The later is implemented as a DLL and called by the main VB program. The two parts are linked by the interactive grammar proposed by Prof. Igarashi. Ms. Hiraga extended the grammar on structure analysis and relationships among structures. She also posed the idea of sub-windows (different analysis goes to different sub-windows). The analysis results can be displayed as different kinds of graphs. Such extensions made the analysis more convenient.
According to this design, I made the detail system design and implementation. Since there was no automatic analysis algorithm at that time, the analysis was performed in a user-specified way. During the developing process, I enhanced the system in the following ways:

1. The initial algorithms of automatic structure analysis and automatic similarity level analysis. The algorithms were not complete then. The structure analysis algorithm did not consider the "Auflikt" condition. The results of similarity level analysis had divergences with the user-specified one. But in any case, the system could do the basic automatic analysis by that time.

2. Improving the interactive grammar environment in various ways. In order to increase error tolerance, a keyword mechanism was implemented. The system will try its best to recognize the command input by the user, even if there are misspellings or wrong word orders. Another utility called next-action-prompt (which prompts the user next available actions after one analysis step) was also implemented. Such improvements helped users to use the system smoothly.

By that time, the first version of DAPHNE system was ready for use.

In 1998, DAPHNE was published in the SIGMUS, and was tried out by musicians. According to the feedbacks, Ms. Hiraga and I realized that analyzing on the actual scores is what users really want. We decided to add actual scores to the system.

Ms. Hiraga’s idea was to generate score images directly from score data (EUROPA data) in real time and display the analysis results on the image. It’s a good idea but had several fatal weaknesses:

- Since the musical information stored in score data files are incomplete, the generated score images are lack of information too;
- Users could hardly accept the “generated” scores because it looks so different with the real ones;
- The user could not do any analyses on this kind of score. The score is only used for displaying results;
- It was very time-consuming to generate score images (about 30 seconds per image).

On the other hand, I tried to use another way to provide a more realistic analysis environment. The basic idea was to use scanned score images. Along with the score data file and the image position file, the user can analyze on the score images directly. Later this method was adopted in DAPHNE. According to Prof. Igarashi’s suggestion, two monitors were used to display more scores simultaneously.

In 1999, Dr. Hiraga finished her doctor program. New master program students joined to the group to do system extensions and implementations. Meanwhile, I began to take charge of the whole system designs. The most important extension to DAPHNE was the algorithm of automatic
analysis of structure functions, tonal, harmony, and musical forms.

I took the responsibility for both structure functions analysis and musical forms, which would be under the directions of musicians.

Mr. Sekiguti and I developed the harmony analysis part, which is not finished yet.

Besides analyses, I also integrated the visualization of performance data, which was developed by Ms. Urushibara, into DAPHNE.

In 2000, my primary tasks were:

➢ The applications of DAPHNE on various aspects.
➢ More extensions according to the feedbacks from users.
➢ The analysis of the accompaniment system's information.

The most important one was the extension of the data fields of the performance data file — UNI. The analysis results can be embedded to the performance file, including not only the MIDI information, but also the music analysis information. Thereby, it becomes much easier to implement the expressive performance.

According to the suggestions from musicians, multi-version scores for one piece were implemented. This utility is for comparing different analyses. Different scores will generate different analysis results and furthermore, different performances.

The following table reveals the history of DAPHNE and the members of our research group. Contributions are indicated in the bracket where "*" shows my works.
<table>
<thead>
<tr>
<th>Time</th>
<th>Design</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1996</td>
<td>Fundamental theory (Igarashi)</td>
<td>LISP system under UNIX (Iyatomi)</td>
</tr>
<tr>
<td></td>
<td>Interactive grammar (Igarashi)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Performance rules (Iyatomi)</td>
<td></td>
</tr>
<tr>
<td>1997</td>
<td>Object technique (Hiraga)</td>
<td>Structure analysis (*)</td>
</tr>
<tr>
<td></td>
<td>Specification (Igarashi, Hiraga, *)</td>
<td>Similarity level analysis (*)</td>
</tr>
<tr>
<td></td>
<td>Algorithm of structure analysis (*)</td>
<td>EUROPA support (*)</td>
</tr>
<tr>
<td></td>
<td>Algorithm of similarity level analysis (*)</td>
<td>Interactive main control window (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analysis sub-windows (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UNI file (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Score image file (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Score description file (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Score image window (*)</td>
</tr>
<tr>
<td>1999</td>
<td>Structure function analysis (*)</td>
<td>Visualization of performance data (Urushibara, *)</td>
</tr>
<tr>
<td></td>
<td>Musical form analysis (*)</td>
<td>Structure function analysis (*)</td>
</tr>
<tr>
<td></td>
<td>Harmony analysis (Sekiguchi, *)</td>
<td>Musical form analysis (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Harmony analysis (Sekiguchi, *)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DAPHNE file (*)</td>
</tr>
<tr>
<td>2000</td>
<td>Apply DAPHNE to applications (*)</td>
<td>Multi-version scores (*)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applications (Koike, Tanaka, *, ...)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analysis of information for accompaniment (*)</td>
</tr>
</tbody>
</table>