Work has begun on a software system to support net-based teaching of core music theory courses at the University of Illinois at Urbana-Champaign (UIUC). The first milestone in the project has been the development of kernel software capable of performing automatic analysis of music. Because of the continuing importance of four-part writing in theory textbooks and coursework, the software is initially being developed and tested using Bach chorales as the primary source material. Automatic chorale analysis includes triad/seventh identification, inversion analysis, nonharmonic tone identification and classification, tonal center identification, cadence classification, functional harmonic analysis, and detection of stylistic anomalies. The ability to analyze different styles of composition will be addressed in future releases of the software. The analysis program described in this article provides the analytical basis for implementing the Music Theory Workbench (MTW), a network-based software environment supporting classroom instruction and individual practice in harmony, counterpoint, and score reading.

Background and Motivation

During the past 20 years, computer applications have had a profound impact on research and pedagogy in the areas of music composition, sound synthesis, recording, and publication. Topics in music theory such as harmony and counterpoint involve formalized concepts that are also amenable to software modeling. I have been teaching core music theory at UIUC since 1995. Through the course of my first semester here, I was struck by some of the changes in higher education that have been gradually occurring over the past decade. My general observations—admittedly non-scientific, taken only from my own experiences and those of my colleagues—are that, relative to UIUC theory instruction 15 years ago: [1] freshmen are now entering the university with less preparation in music theory, [2] the overall teaching load on professors has increased, and [3] the skill of teaching assistants is more variable. Because UIUC is a major public university, it is reasonable to assume that similar trends exist at other institutions of higher education, and that these trends reflect some broad, underlying economic and social conditions that are likely to persist into the future. At the same time, during the last 15 years, advances in computer technology have proceeded at an astounding pace, to the point where most universities and high schools provide personal computers and access to the Internet. However, despite these changes, theory classes here (and elsewhere) are structured basically the same as they were 100 years ago. I believe that current conditions argue strongly for a new mode of music theory instruction, one that offers a response to existing economic and educational challenges, while at the same time deepening the educational experience of the student and broadening the role of the teacher. This is the motivation behind the Music Theory Workbench project: to implement a network-based learning environment that unites music theory pedagogy and distance learning. By its very nature, a technologically sophisticated system that embeds music theory instruction into the network could have an enormous impact on how music theory is taught, who benefits from it, and by how much.
The study of harmony and counterpoint lie at the center of most theory programs. As any theory instructor can relate, these subjects are difficult to teach and the preparation and grading of assignments demands considerable time and energy on the part of professors and teaching assistants. University-level harmony courses typically require students to demonstrate their understanding through the composition of short four-part musical studies that are critiqued primarily on how well they exhibit the normative behaviors of classical harmonic and voice-leading practices. Owing in part to the inherent complexity of this subject matter, and in part to the multitude of possible "solutions," these compositions are notoriously time consuming to review and correct. The labor-intensive grading process, in turn, limits the amount of work a student can submit for correction and feedback. This limitation grows in proportion to the size of the class, and in core music theory courses (my lectures have 50–120 students; the sections 20–35), it limits the depth of understanding and the familiarity with materials that a student might otherwise achieve.

The overall goal of this project is to create a software environment that automates what is currently only possible by human interaction. In addition to removing the limitation on how much graded practice a student can perform, the MTW will also improve education in theory by providing automatic test and homework generation; a consistent level of grading expertise; on-line examples and analytical databases; interactive, playable class examples and hypertext course documentation; and a vehicle for teachers to "market" their courses to wider audiences.

When completed, the MTW will offer a powerful set of tools for teaching harmony, counterpoint, and score reading. The software will consist of client and server components, as shown in Figure 1. The client side will be defined by a graphical "music theory editor" on the user's machine. This editor will allow a student to display, load, save, edit, and play back musical material. The server side will consist of the analytical kernel, communication software, databases of examples, and on-line materials related to specific music theory courses. Taken together, the client and server sides of the MTW will provide a distributed learning environment flexible enough to support the challenging and creative types of assignments that typify upper-level music theory courses, and at the same time will demonstrate enough "intelligence" to provide instructional expertise in the form of mu-
rical problem detection, automatic assignment correction and grading, automatic test and exercise generation, and analytical searching of digitally encoded music.

The ability to analyze music lies at the core of this system, and it provides the motivation for the work described in this article. Four-part harmony has been selected as the first priority of the project, because it remains a central pillar of most harmony courses and is an essential component of theory textbooks such as *Harmony and Voice Leading* [Aldwell and Schlachter 1989], *Harmonic Practice in Tonal Music* [Gauldin 1997], and *The Elements of Music* [Turek 1996]. For this reason, the chorales of J. S. Bach are used as the primary test material in the first phase of development. Bach chorales were chosen because they provide short yet extremely complex examples that present a variety of analytical challenges. They will also serve as important source material for future system capabilities such as automatic exercise generation and analytical searching (finding examples of specified harmonic or melodic patterns).

Although four-part texture is the project’s current focus, the software is being designed to allow other musical styles to be addressed in future development cycles. As subsequent sections of the article explain, an analysis does not proceed directly from a score: compositions are first parsed for information that is then stored in a canonical form for analysis. This extra step provides a level of abstraction above the actual score data, and means that the information on which an analysis proceeds could potentially come from a variety of sources. Moreover, the analytical descriptions can serve a pedagogical function if they are used as “templates” for music generation. Given a template, it is not difficult to provide a “reverse parsing,” i.e., to generate specific musical instantiations from the more general description. Templates could be provided by the teacher, computed, or obtained by the system from the analysis of a piece. Regardless of origin, templates could serve as a means for creating variations of the same basic exercise but tailored to a particular pedagogical goal or skill level.

**Comparison to Related Research**

There have been a number of research projects related to automating various aspects of music theory. Many projects that demonstrate theoretical expertise are primarily concerned with computing stylistically correct compositions. William Schottstaedt’s automatic counterpoint program [Schottstaedt 1989], David Cope’s Emmy system [Cope 1991], K. Ebcioglu’s CHORAL expert system [Ebcioglu 1992], and HARMONET [Hild, Feulner, and Menzel 1991] are examples of this category.

Several projects have also attempted to model the acquisition of analytical/stylistic information from musical scores using neural networks. Hörnel and Ragg [1996] use a neural net to learn musical structure and style from musical input. Winograd [1968] performs harmonic analysis of Bach chorales by defining a formalized grammar of tonal music and finding a “best” harmonic account according to the grammar. Scores are parsed from back to front, in order to limit the search space. The system does a good job of harmonic analysis, but it does not consider the linear aspects (i.e., attempt to classify melodic relations), nor does it detect anomalies. H. John Maxwell’s expert system for harmonizing analysis [Maxwell 1992] addresses many of the same issues faced in this project, and there is a considerable overlap of results. Both systems analyze musical input for harmonic functions, and both perform chord classification and nonharmonic tone analysis as subtasks in the process. Because of the different motivations underlying each project, the systems differ in their basic approach to the problem as well as in their specific implementation strategies. Maxwell’s elegant system is intended (in part) to model the intuitive judgments of a human analyst. The analyst’s judgments are explicitly modeled as production rules (“if/then” statements) that are grouped according to analytical task. An analysis is produced by “forward chaining” rule sets. “If” rule conditions are tested, and if true they trigger “then” actions that assert changes into the knowledge base. Rules in a rule set stop firing when all implications have been processed. To perform a functional analysis, the system first identifies chords, then judges an overall key based on the last
and first chords uncovered in the analysis, and then establishes an optimal tonal context and harmonic function for each sonority based on heuristic judgments that affect numerical confidence factors.

In contrast to the expert system approach, the MTW system is not intended to represent the judgments of a human expert, and the analysis is approached from a different perspective. In this system, musical data are matched against models to determine classification and outcomes. As opposed to the expert system approach, the process does not involve heuristic judgment, nor does it proceed in a top-down manner based on a determination of global key areas from starting and ending sonorities. In fact, the notion of key is not relevant to the process. Instead, the MTW first searches everywhere in a composition for local tonal center definitions (dominance of a pitch regardless of mode). A map of these points is created, and then a bottom-up analysis attempts to generalize between adjacent points of local certainty.

### The Analysis

The analytical process of four-part music is designed as a series of steps, called passes, that take musical data as input and generate an analytical graphic score. A representation of the overall control flow is shown in Figure 2.

The user may directly control many aspects of the analytical process. The most basic level of control is through specification of the models on which the system is based. Models declare chord patterns, nonharmonic tone signatures, tonal center definitions, modulation paths, and style checks. The ability to change the theoretical structures on which the system is based has obvious utility for a pedagogical system. Figure 3 demonstrates how the addition of new harmonic models might be useful in the context of teaching extended tonal techniques.

It is hoped that by using declarative models and
modularizing the analytical steps, the current system can evolve into a more general analytical search engine that supports atonal and set analysis as well as tonal harmony. If the analytical steps were refined into a more formalized set of generic procedures, each type of analysis could then define methods on the generic routines that customize the process as appropriate. If successful, a single analysis protocol might be applicable to a variety of different theoretical tasks and musical styles.

Pass 1: Parsing and Sonority Analysis

The first step in the analysis is to parse the input score data into a time line of structures that hold all the information necessary to the subsequent analytical steps. These structures, called **sonorities**, represent the set of all sounding notes that are present whenever any note is articulated in the composition. When a sonority is parsed from the score, its metric position and stress level are determined and its notes are reduced to an interval set (with octave displacements and duplications eliminated). This interval set is then matched against a table of allowable models to determine if it is classifiable. Note that these models could be used to classify either harmonic or melodic figures. For four-part music, the matching table contains models for all triad and seventh chords, plus some incomplete models that occur with regularity in tonal music, such as triads and sevenths without the fifths. The matching entry in the table produces the classification type of the sonority in question. A classified sonority is then further analyzed for inversion, doubling, and voicing information. As sonorities are parsed from the score, they are linked together to form an analytical time line. The link is established using a structure called a **transition**. Transitions model information about the motion between sonorities, such as the intervallic distances between chord roots and between bass tones.

Pass 2: Nonharmonic Tone (NHT) Analysis

Sonorities that were not classified during Pass 1 are subjected to a nonharmonic tone analysis in Pass 2. This process must first identify which tones belong to an underlying harmony and which do not belong. McHose defines a nonharmonic tone as “a tone which appears in a vertical sonority but plays no part in the theory of inversion of that sonority. The nonharmonic tone does not have the spelling of any of the intervals that make up the implied harmony. The nonharmonic tone is related melodically to one of the members of the chord” [McHose 1947]. Metrical stress also influences the interpretation of nonharmonic tones. Because our rhythmic system is proportional, “weak” and “strong” designation is appropriate at multiple mensural levels: for example, measure vs. beat as well as beat subdivision. For mensurally weak sonorities, the “preparation” (the closest metrically stronger sonority to the left) serves as the “target” for identifying nonharmonic tones in the unclassified sonority. Each tone in the unclassified sonority is tested to see if (1) it is in the target’s set of harmonic tones; (2) it complements the target’s set [i.e., fills in incomplete harmony]; or (3) it is already accounted for as a nonharmonic tone in some other figure. If none of these conditions is true, the tone is identified as a nonharmonic tone in the unclassified sonority.

Nonharmonic tone identification for strong sonorities is a more difficult task. The analysis must make two independent tests, one for “momentary” tones [a tone that does not appear in the previous sonority and does not last a full beat], and one for “stale” tones [tones that were consonant in the last sonority, but are dissonant in the current sonority]. It is possible to have both conditions in a single sonority; for example, an accented passing tone in one voice and a suspension in another. For strong unclassified sonorities, the target sonority is the next-weaker sonority “to the right.” Successor tones [the equivalent tones in the target] are substituted into the current sonority, and if a chord model can be recognized, the tones that were replaced by the substitution are identified as nonharmonic tones. The analysis procedures in Pass 2 can also be used to identify misplaced NHTs. An optional analysis parameter defeats the check for strong/weak compliance in the target selection. This allows “mistakes” to be found; for
example, “suspensions” that are prepared and resolved on strong positions relative to the dissonance, a typical error that a beginning theory student might make, are readily identified.

**NHT Classification**

Once the nonharmonic tones of a sonority have been identified, they are classified in a process similar to Pass 1. The preparation, NHT, and resolution tones are reduced to their interval models, and this melodic “signature” is matched against a table of NHT models to determine membership. The system currently identifies upward and downward variants of the following nonharmonic tone types: passing tones (PT), neighbor tones (NT), changing tones (CT), anticipations (ANT), incomplete neighbor tones (IN), and suspensions and ritardations ($7\rightarrow6, 4\rightarrow3, 2\rightarrow1, 7\rightarrow8$). Figure 4 shows the detection of multiple and consecutive nonharmonic tones in close proximity.

**Seventh-Tone Classification**

Seventh tones have a dual nature. On the one hand, they must be recognized as belonging to the basic harmonic vocabulary of tonal music. On the other hand, they are dissonant with respect to the root and are often handled as NHTs in voice leading. For this reason the system also performs NHT classification on all seventh tones. This information is used in the exception analysis to point out nonstandard usage (for example, places where the seventh tone of a $V^7$ does not resolve downward by step).

**Consolidation**

The analysis currently classifies any possible chord sonority as a triad or seventh. But in some cases, a sonority that can be recognized as a chord is more appropriately explained by melodic principles. During the next phase of development, a consolidation process will be implemented to address these and other related issues. Consolidation means that a detail at one level of analysis can be explained or “subsumed” by a more general or a higher-level structure. A consolidation process will permit control over the “granularity” of analysis, i.e., how much detail should be considered relevant to the determination of some particular feature. This in turn will permit the analytical routines to work with structural units above the individual sonorities that make up the surface flow of harmonic changes. The consolidation process could use factors such as mensural level and metric weight, agogic stress, strength of progression, and location in the cadential cycle as indicators of the need to group local sonorities into larger formal units. This consolidation process can also be used to identify pedal tones, and to account for chords that are the result of neighbor relations in one or more voices. By applying it to higher and higher levels of analysis, harmonic rhythm and tonal center evolution could also be processed.

Figure 5 shows the results of the sonority and nonharmonic tone analysis on the first half of Chorale 126 [Riemenschneider 1941]. This example will be used again to show the results of the tonal and exception analysis.

**Pass 3: Tonal Analysis**

By the end of Pass 2, the sonorities and nonharmonic tones have been classified. Any remaining unclassified sonorities become “invisible” to the subsequent harmonic analysis. (The analysis currently identifies every sonority in the test suite of ten chorales.) The next step in the analysis is to...
construct a functional theory that explains the role of each sonority in the tonal evolution of the piece.

**Cadence-Point Identification**

Cadence points are critical indicators of underlying harmonic organization in tonal music. The system currently analyzes a composition only for structural cadence points. A structural cadence point is a resting point in the composition made explicit by the composer. For chorale-style compositions, this includes section endings and any sonority marked by a fermata. Structural cadence identification does not take the place of a more general cadence analysis, but since it is the more basic feature, it was selected to be implemented in the first phase of development.

**Tonal Center Tones**

To identify tonal centers in the composition, the chord models and root movements of all sonorities are searched for signatures that point to the precedence of some tone over others. These tonal center tones are identified using two independent criteria. First, a tonal center tone is touched if it is the root of a major or minor chord preceded by either a major chord a perfect fifth above or by a diminished chord a minor second below. (Touching finds false positives, for example, in a I→IV progression, the IV will be marked as touched.) Second, a tonal center tone is implied by a dominant seventh or any sort of diminished chord. A dominant seventh implies a tonal center tone a perfect fifth below the root of the chord; a diminished chord implies a center one half-step above the root of the chord.

Using these two criteria, the system pairs each sonority with its closest successor that changes root, and tests for confirmation of the tonal center. For chorales, a tonal center is confirmed by

- A tonal center tone being implied and then touched, i.e., \( V^7 \rightarrow \{I, i, VI, \text{or} vi\} \) or \( \{\text{vii}^6 \text{or vii}^7\} \rightarrow \{I \text{or} i\} \)
- A touched structural cadence, i.e. \( V \rightarrow \{I \text{or} i\} \), with I a structural cadence point; or
- A touched successor from a structural cadence, i.e. \( V \rightarrow \{I \text{or} i\} \) with V being a structural cadence point

Note that in the case of a dominant seventh chord, the center will be confirmed even if the root movement proceeds by step in a “deceptive” motion. The identification process may seem overly sensitive in that it finds tones that are not really tonal centers in the composition. These false positives are dealt with later, through a process of center consolidation. Another apparent problem is that what might, in fact, turn out to be \( V \rightarrow I \) harmonic motion will not in and of itself confirm a tonal center unless one of the sonorities is a struc-
tural cadence. However, this does not mean that the progression will not be found, only that the fall of a fifth in and of itself is too weak to confirm the establishment of a tonal center. (The motion could be \( V \rightarrow I, I \rightarrow IV, VII \rightarrow III, \) and so on.)

**Tonal Centers**

The system next establishes tonal center descriptions for every tonal center tone uncovered in the search. Also included are “theoretical centers” that the analysis posits a minor third below and above the actual tonal center tones that were found. This behavior can be controlled by an analysis parameter that allows the use of (1) all possible tonal centers, (2) only the set of centers derived directly from the analysis, and (3) a set of centers specified directly by the user. The number of tonal centers the system considers can affect the “granularity” of the tonal analysis; the default configuration represents a middle ground given the possible configuration choices.

The tonal center descriptions contain statistical information about each of the centers. This includes the number of times and positions in the composition the tonal center was implicated, touched, and confirmed, the interval relationship to the other centers (providing evidence of how tonal centers are related), and the agogic weight of the center’s tone (i.e., the amount of time the confirmation tone occupied during the composition). Once the set of tonal center descriptions has been gathered, it is sorted and ranked according to the number of times each center was confirmed in the composition. The number of confirmations provides a strong indication of the relative importance of a tonal center in the composition, but it does not suggest an “overall key” of the piece. Instead, the system uses the tonal center confirmations as a map to provide the skeleton for elaborating a functional analysis.

**Tonal Theory Lines**

Once all tonal centers in the composition have been identified, the system computes a tonal theory line for each of the tonal center descriptions. A tonal theory line details the possible functional role of every sonority in the composition in the context of that theory’s tonal center. The analysis maintains a full functional theory for all the tonal centers it considered. This means that the system can provide all possible accountings for any region of interest, a feature that could be exploited for pedagogical purposes.

**The Tonal Center Model**

In order to determine whether a given sonority can be included in a tonal theory line, its chord model is matched against a tonal center model. The tonal center model is a declaration of the chord models and degrees that define a tonal center. The easiest way to visualize this model is as a two-dimensional cube in which the columns (the horizontal dimension) define the set of intervals that constitute “scale degrees” for the center, and the different rows (the vertical dimension) hold the variant chord models on each scale degree. The “mode” to which any particular chord model belongs can be determined from the index of the row in which the variant is placed. Any number of rows (modes) can be added. For example, rows could be provided for major, octatonic, dorian, mixolydian, melodic minor, and so forth. An analysis-control parameter allows matching to be constrained to just a subset of the possible modal rows. The default behavior is to match against all variant models.

To determine whether a particular sonority can be included in a tonal center’s theory line, the interval distance between the root tone of the sonority and the tonal center’s tone is first calculated. This effectively transposes the tonal center model to a particular tonal center note. However, transposition in this case is not just the number of half-steps. All interval measurements in the system account for spelling (e.g., C to C-sharp is not the same interval as C to D-flat, although they are both a semitone apart). Intervals between two notes are represented as formatted integers \( LLSS \), where \( LL \) is the distance between letter names and \( SS \) is the number of half-steps spanned. The system declares constants for all intervals from doubly diminished to doubly augmented up to the...
octave (by default, the interval routines reduce compound intervals). Once the interval distance is measured, it becomes a candidate “scale step” (column) in the model. If that step is actually defined in the tonal center model, the chord template of the candidate is checked against the row variants at that step position. If the template is found, then the system can account for the sonority as a functional symbol in that tonal theory line. The functional symbol itself is easily calculated from the identified column index (scale degree) and variant chord model. The model is also useful for following chord relationships very quickly. For example, referencing V of V of III from a given tonal center requires two table lookups. Figure 6 provides the content of the basic tonal center model.

**Functional Analysis**

The outcome of the tonal center identification process is a set of tonal theory lines, one for each tonal center included in the analysis. Each line contains the interpretation of every sonority in the composition given the line’s tonal center. The system next provides a functional analysis, namely, a description of the evolution of tonal centers over the course of the composition. As a preliminary step in this process, any tonal theory line that does not play a significant role in center confirmation is removed from the analysis. A center’s significance is determined through a process of subsumption: a tonal theory line is eliminated if it contains no confirmations and if all of its functional symbols can be explained in terms of centers that do have confirmations. The result is a set of tonal theory lines that either contain confirmations or explain sonorities that would otherwise not be accounted for by any other theory.

A functional analysis in this system simply refers to the determination of the tonal center or centers active at each position in the time line. To accomplish this, the locations of every tonal center confirmation are first gathered. Since these points are unambiguous indicators of tonal center establishment (however brief), the total map serves as a framework for bottom-up elaboration of the sonority-by-sonority tonal evolution in the composition. To compute the functional analysis, the program attempts to “fill in” between pairs of points in the confirmation map. This means that the confirmation map is processed two points at a time—a starting (left-side) tonal center, and an ending (right-side) tonal center. (The final sonority in the piece may or may not be a confirmation, and the initial sonority is never a confirmation.) Once analyzed, the old left side becomes the right side, and a new left side is read from the confirmation map (this overall direction is not important).

The section of music between confirmation points is processed according to a stepwise process. Each of the points defines a “region of explanation.” The goal of the analysis is to extend the starting center’s region forward and the ending center’s region backward as far as possible and as long as the sonorities encountered are contained in the relevant center’s tonal center model. The effect of this process is to extend areas of relative
certainty (the confirmation points) as much as possible in both directions. If the left side and right side are the same tonal center, and there are no unexplained sonorities, then that center and its functional symbols are used for the entire region. If, however, the left- and right-side centers are not the same, then one of three situations can occur: (1) the different tonal center regions cross or meet, defining a shared region; (2) the different tonal center regions adjoin; or (3) the different tonal center regions fail to adjoin. If the starting and ending regions cross or meet (situation 1, where the ending of the left-side region overlaps the beginning of the right-side region by at least one common chord), an area of shared harmonic functionality is identified. The system marks this as a “shared harmonic region,” and then selects the latest chord from the region as a “pivot” chord from the first tonal center to the second. (Determining a good pivot chord is in reality more complex than this, and the next release of the software will analyze the shared region to select the most appropriate chord.) If the tonal center regions do not overlap, then either they adjoin (situation 2), in which case the tonicization of the right-side region does not involve a pivot chord, or else neither region explains all the harmonic activity between the two points (situation 3).

There are various ways to approach this third situation. The system currently uses a “modulation model” to help resolve common situations. The modulation model defines chord models and scale degrees not contained in the tonal center model that might indicate a shift in tonal centers. Each variant in the modulation model is represented by a chord model and the tonal center tone it suggests. For example, the tonal center model for Bach chorales does not contain a major III chord, and its entry in the modulation model suggests a possible V/ vi → vi. The system matches unexplained edges in the analysis to entries in the modulation model to determine whether an edge chord can be accounted for as a possible modulation. If a match is found, the left- and right-theory regions are re-evaluated in terms of the newly implied tonal center to see if the activity can be more fully explained. If so, then the region is filled in using the new tonal center; if not, then the sonorities in the gap are marked as “undetermined” in the tonal analysis. With one exception explained later, the tonal center and modulation models account for all ten chorales in the test suite. Many of the test chorales contain complex harmonic passages, an example of which is provided in Figure 7.

Pass 4: Exception Analysis

By the end of Pass 3, all the sonorities in the composition have been analyzed for their chord type, inversion, voicing, and tonal function. The last step in the process is to perform an “exception analysis,” i.e., to identify stylistic anomalies and errors. Although exception analysis is a crucial component in a system that will support music theory pedagogy, the nature of an exception and how it should be interpreted depends to a large degree on the judgment of the user. For this reason, exception analysis has been designed with flexibility in mind; users have full control over which exception conditions are tested and how they are ranked. In future releases, users will be also able to add new exception checks to the process. Exceptions are annotated in the analytical score by a colored circle drawn around the relevant note head(s). A table printed at the start of the analysis explains every exception identified in the score.

Severity Levels

Each exception has an associated severity level that indicates its relative importance. The system provides four severity levels [ranked from zero to three] that denote the conditions: “ignored,” “notice,” “warning,” and “error.” An ignored exception [level zero] does not appear in the analytical results. Notices [level one] are intended to convey information about a particular feature of the music. A warning [level two] is intended to highlight a feature that might be problematic. An error [level three] indicates a problem that should be corrected. Each severity level is graphically distinguished by a unique color in the analytical score so a student can quickly ascertain both the num-
Exception Types

Exceptions are classified according to their type. The system identifies three broad exception classes: motion exceptions, voicing exceptions, and harmonic exceptions. Within each class, a number of subtypes are defined. Each exception class has an associated default severity level. Table 1 summarizes the classes, subtypes, severity levels, and interpretations of the exceptions currently recognized by the system. Many other categories and subtypes are certainly possible, and more types will be added in the future.

Analysis Output

Once the exception analysis is complete, the results of the analysis are saved as an EPS (encapsulated PostScript) file containing an annotated graphic score. The score includes summary information about the analysis, as shown in Figure 8. The EPS file is produced using Common Music Notation (CMN), a music-notation language developed by William Schottstaedt at CCRMA, Stanford University [Schottstaedt 1992]. The graphical presentation uses the following conventions:

- Colored note heads indicate different analytical distinctions: black for harmonic tones, blue for nonharmonic tones, and red for unclassified tones.
- Colored circles around note heads indicate exceptions. The color of the circle denotes its severity level: red for errors, purple for warnings, and green for notices.
- Figured bass symbols underneath each sonority provide inversion analysis; nonharmonic tone classifications are printed in blue.
- Functional symbols under a sonority define its role in the current tonal center or centers. Centers are represented as labeled rows.

The analytical output also includes summary information: a tonal center breakdown (for each tonal center uncovered in the analysis, its mode,
definitional characteristics, and agogic weight), an automatic modal analysis of the chorale melody (many Bach chorales are not major or minor), and an exception table explaining each exception found in the music. This output for the first half of Chorale 126 is shown in Figure 8.

To produce the graphic score, the program computes a CMN expression (a Lisp expression) that describes the presentation to be rendered to the PostScript output file; then, CMN determines the proper layout and generates the graphic image. (All the musical examples in this article were produced using CMN.)

### Analysis Results

During this first phase of development, the software is being tested on a collection of 10 Bach chorales: numbers 1, 30, 113, 126, 140, 173, 216, 217, 262, and 334 in the Riemenschneider edition. Chorales were selected that provide particular analytical challenges. The set includes harmonization of modal chorales, chromatic bass lines, chromatic modulations, ambiguous tonal regions, simultaneous and consecutive nonharmonic tones, and suspensions with moving bass lines. Also included are examples of exception such as direct perfect intervals, voice ranges and spacing, and voice overlap and crossing. The analyses of the chorales in the test suite are essentially correct. It is not possible to provide these results in the space of this article, but the full test suite is available via the World Wide Web at http://www-camil.music.uiuc.edu/Software/mtw/ata/examples. Each chorale in the on-line catalogue is represented by its analysis output file (PDF and PostScript formats) and a MIDI file. Over the course of this year, I hope to provide analyses of many more chorales from the Riemenschneider edition.

As mentioned in the section on consolidation, the most global problem with the current system is that too much detail is presented. This will be addressed by implementing consolidation. Another unresolved issue relates to the minor dominant and subtonic chords. Since these are

<table>
<thead>
<tr>
<th>Exception Class</th>
<th>Severity</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motion exceptions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parallel octaves</td>
<td>3</td>
<td>Consecutive octaves, same two voices</td>
</tr>
<tr>
<td>Direct octaves</td>
<td>2</td>
<td>Octaves by similar leap, any two voices</td>
</tr>
<tr>
<td>Parallel fifths</td>
<td>3</td>
<td>Consecutive fifths, any two voices</td>
</tr>
<tr>
<td>Direct fifths</td>
<td>2</td>
<td>Fifths by similar leap, any two voices</td>
</tr>
<tr>
<td>Parallel unisons</td>
<td>3</td>
<td>Consecutive unisons, any two voices</td>
</tr>
<tr>
<td>Direct unisons</td>
<td>2</td>
<td>Unisons by similar leap, any two voices</td>
</tr>
<tr>
<td>Diminished skip</td>
<td>1</td>
<td>Diminished melodic interval, any voice</td>
</tr>
<tr>
<td>Augmented skip</td>
<td>1</td>
<td>Augmented melodic interval, any voice</td>
</tr>
<tr>
<td>Nontriadic skips</td>
<td>2</td>
<td>Consecutive skips not triad, any voice</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Voicing exceptions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Voice crossing</td>
<td>1</td>
<td>Voice above/below adjacent voice, any pair</td>
</tr>
<tr>
<td>Voice overlap</td>
<td>1</td>
<td>Voice above/below previous adjacent note</td>
</tr>
<tr>
<td>Voice range</td>
<td>2</td>
<td>Voice out of normal vocal range</td>
</tr>
<tr>
<td>Voice spacing</td>
<td>1</td>
<td>Wider than octave, upper voices</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Harmonic exceptions</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Missing third</td>
<td>2</td>
<td>Sonority missing third of chord</td>
</tr>
<tr>
<td>Retrogression</td>
<td>0</td>
<td>Harmonic retrogression (V–IV, etc.)</td>
</tr>
<tr>
<td>Leading unresolved</td>
<td>3</td>
<td>Leading tone unresolved, outer voices</td>
</tr>
<tr>
<td>Seventh unresolved</td>
<td>3</td>
<td>Seventh tone of V not properly resolved</td>
</tr>
</tbody>
</table>
extremely weak tonal center indicators, the software currently allows the user to choose how these should be interpreted in the tonal center model. The system should instead base this decision on the actual musical context. Cadence analysis and better pivot-chord selection are also needed. A few Bach chorales contain suspensions with ornamental resolutions in which the NHT does not proceed directly to the resolution. The system currently searches for stepwise ornamental resolutions in suspensions without using the NHT models, and this should be corrected. Other ornamental suspensions, such as those defined by a leap away from and back to the tone of resolution, are currently not identified. Pedal tones are also currently not found; this limitation will be addressed by consolidation.

Es ist Genug is the most complex chorale in the test suite [and one of the most complex in the entire Riemenschneider edition], and it contains the only passage on which the system performs poorly. Most of this chorale is actually analyzed quite well, including the famous whole-tone opening phrase. The breakdown in the analysis occurs starting on

"Durch Adams Fall ist ganz verderbt" (#126)
Automatic Analysis by the Music Theory Workbench
CMN Postscript output of 7-Mar-99 8:21:47.

Chorale melodic mode: a minor

Tonal Center Analysis:

<table>
<thead>
<tr>
<th>center confirmed</th>
<th>implied</th>
<th>reached</th>
<th>ticks</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>a</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Exceptions:

5: Diminished melodic interval, bass voice.
12: Voice overlap, alto above previous soprano.
12: Direct fifths, tenor and soprano.
22: Diminished melodic interval, tenor voice.

Downloaded from http://www.mitpressjournals.org/doi/pdfplus/10.1162/014892699559977 by guest on 02 May 2021
the last beat of measure 15. This chord (G-B-F) is best explained by voice leading, but sounds as if it were a German sixth in B minor. However, because of its spelling, it cannot be classified as an augmented sixth chord (the F would have to be written as E♯), and the system cannot account for it in any of its tonal lines. After this point, a harmonic context cannot be re-established until the fourth beat of the next measure, where the D♯ diminished seventh causes the system to analyze measure 17 as a retrogression in E (vii° → IV). However, this passage really functions as a “slipped modulation,” i.e., it would best be accounted for in A (vii°/V → I°). This harmonic signature happens enough in tonal music that the system should check for it during tonal center analysis, but it currently does not. Correcting the analysis of the three-chord chromatic descent is more problematic. My current preference is to search for root movements involving fifth relations, and use the tonal center model to prefer solutions that allow closely related centers to be “chained together” in as few steps as possible that account for the motion. In this way, the system could account for the passage by following the F-sharp–B–E–A chain or roots that underpin the chromatic passage.

Software Implementation

The MTW is implemented in Common Lisp. Analysis routines are generic functions, and the score data (voices, parts, note events, key signatures, and so on) are represented in Common Music, a software environment for computer-based music composition (Taube 1997). Common Music is used to load, save, edit, and perform the musical data. The analytical software and models described in this system are separate from the host system. Since the analysis software will eventually be used to support an interactive theory editor, it has been designed with speed and language portability in mind.

As mentioned in an earlier section, Pass 1 parses external score data into a canonical representation. This representation can be manipulated efficiently, because the musical data (notes, chords, rhythms) have been “digested” and encoded into formatted integers. The analytical operations that are performed from this point on basically involve fast bitwise operations. Matching is implemented using hash tables and table lookups. The result is a fast analysis process—on a 266 MHz PowerPC, the full suite of 10 Bach chorales is completely analyzed in less than 1 sec, including the parsing process.

The eventual client and server sides of the project face different implementation issues. The client will almost certainly be implemented in Java, but the choice of language in which to implement the server is still undecided. My preference is to keep the system running in Common Lisp, because development is so easy in this environment. John C. Mallery at the Massachusetts Institute of Technology has implemented an HTTP server in Common Lisp (Mallery 1994) that could provide the communication services needed to implement the server side of the Music Theory Workbench.

Future Directions

A significant amount of work on the analytical kernel has already been completed. However, the overall project is still in its infancy. Future directions include near-term, intermediate, and long-range goals. The analytical kernel remains at the center of development for the near future. Short-term goals include strengthening analytical passes (better pivot selection, etc.), designing and implementing a consolidation pass, adding additional exception checks, and combining the tonal center and modulation models into one large chromatic model. Intermediate goals include implementing harmonic and voice-leading templates—the analytical descriptions that will serve as the basis for automatic homework and test generation. Another intermediate goal is to test the automatic analyses on the entire set of 371 chorales in the Riemenschneider edition, and to make the results available on the Internet for those interested in tracking the project. These will serve as snapshots of the abilities and deficiencies of the analytical software in its development cycle. Longer-term goals for the project include extending the analyti-
cal kernel to include other compositional forms and styles, designing the client and server software, and implementing species-counterpoint analysis.

References


