Priming Computer-Assisted Music Composition through Design of Human/Computer Interaction

MICHAEL HAMMAN
1814 Larch Place
Urbana, Il 61801
United States

Abstract: - This paper considers recent developments in the psychology of creativity and human/computer interaction, in order to advocate that the computer be understood not merely as a tool for building tools, but as a tool for constructing conceptual frames by which we come to understand and extend problem domains. Such an understanding of the computer allows for evaluation of its use in the creation of progressive art. Use of the computer in music composition and sound synthesis has a long history (more than 40 years) and provides a wealth of theoretical and experimental data on creative idea generation and human/computer interaction. Recent theories in cognitive psychology indicate that creative thinking involves a process of iterative activation of "cues" which help direct the generation of ideas and the search for problem solutions. Along the way, however, thinking can become *fixated* on particular cues, thus leading to mental blocking and failure to creatively solve a given problem or formulate a new idea. Such a deadlock can be broken by shifting the conceptual framework through which cognitive patterns are engendered. The computer can be understood as a tool for designing interfaces and, as such, as a tool for constructing conceptual frameworks. In this capacity, computers assist not only problem solving, but in designing the very conceptual means by which problems might be formulated. It is precisely this capacity of the computer—the capacity for constructing representations of problem domains and spaces—that compels its use in creative activity like music composition. Several music composition software systems are briefly discussed in light of these considerations.

Key-Words: - Composition theory, cognitive psychology, human/computer interaction, music composition, computer music, computer interfaces, creativity, problem solving.

1 Introduction

Use of the computer in music composition is becoming increasingly common. Software tools for various forms of musical activity proliferate, allowing composers to accomplish everything from editing digitally stored sounds, to synthesizing and designing sound, to constructing algorithms for generating musical patterns. Software tools range from those used for commonplace tasks (e.g. recording and playing media files) to those used for experimental research (e.g. elaborate systems for composing individual sounds to entire musical works). Most tools have particular commercial interests at heart. What all tools have in common is that they frame the manner in which the tool-user comes to understand and internally represent musicals tasks and the larger musical task domain with respect to which those tasks conceptually cohere.

Different cognitive, social, and political conditions are brought to bare when considering the impact of computer technology on artistic research and production, and vice verse [1]. The current paper considers the problem, first from a cognitive psychological perspective, and then from the perspective of human/computer interaction, in order to build a theoretical base for the analysis of particular music composition software systems.

2 Fixation and Incubation in Memory and Creative Thinking

Cognitive psychologist Steven M. Smith has done important research in the cognitive patterns of memory and creative thinking. His theoretical and experimental research accumulates a great deal of preceding work and forms a useful foundation for analyzing the particular kinds of problems that computers might best solve. Smith's theoretical ideas and experimental results will thus be considered in some detail. Of particular interest is his research in what are termed *fixation* and *incubation* in memory and creative thinking, as detailed in his essay of 1995 [2].

2.1 Creative Thinking

Creative thinking requires the construction of *plans*. Plans govern the retrieval of information and production of “cues” determining direction and orientation of the thinking process. From the point of view of cognitive psychology, such plans can be retrieved in one of two ways.

A plan might be retrieved as a *unit* in which case the thought process is reproductive rather than creative. An example might be the process by which one obtains degrees radian from a sin function. Though a solution to a particular problem may not be known ahead of time, the process by which it is arrived at is invariant.

By contrast, a plan might be constructed solely from a set of known elements. In this case, the plan is not fully laid out in advance. As a result, one has flexibility in dealing with new problems or unfamiliar problems that the fully laid out plan cannot provide.

The search of memory within a plan follows an iterative process in which memory is both *probed* and *monitored*. A probe constitutes an assembly of informational units, or elements, that are used to search memory. Each such element acts as a *cue*, activating long-term memory.

Information that is activated is monitored and analyzed: elements that satisfy constraints contained within the initial plan are added to a succeeding probe. Whenever activated knowledge satisfies the constraints defined within a plan, that plan becomes further stabilized.

3.3 Implicit Fixation

The particular elements and memory cues that are included within a probe are determined by recent experiences, long-term memory and permanent knowledge, recently *primed* memory (that is memory that has been recently activated), and contextual information.

Cues can either be intentionally or implicitly included within a probe. Intentional cues are those that are explicitly declared as agents governing a search. For instance, in an anagram problem, one might be presented with a word whose configuration of letters suggests a possible solution. This “hint” at the solution becomes a directing agent in subsequent searches.

Intentional cues are articulated agents within a probe: they can be freely selected or freely discarded. However, once included within a probe, intentional cues can become *implicit* cues. Implicit cues lose the selectivity aspect of intentional cues: being “implicit”, they become largely unconscious. An implicit cue leaves its trace within memory, thus biasing subsequent search. Since they are not conscious agents, implicit cues cannot be withdrawn.

Implicit cues can prime memory in a beneficial way, leading to the relatively effortless solution of problems. This is because implicit cues require very small cognitive effort to maintain, freeing our attention for more demanding tasks. However, if information elements in memory that lead to dead-ends are repetitively primed (through implicit cues), the success of the search is compromised. Thus, while implicit cues reduce cognitive effort, and can greatly assist in a successful search, they can also lead to *fixation*: an impasse in the ability to backtrack from a dead-end. Such a blockage is termed an “implicit assumption.” Implicit assumptions prevent formulation of new ideas and debilitate problem solving strategies.

In tasks involving creative problem solving and idea generation, it is sometimes necessary to take extreme measures to “unblock” oneself from the grips of an implicit assumption. More often, one need only take a break and engage in some unrelated activity, in order to refresh memory.

3.4 Fixation as a Block to Memory

A number of studies have been conducted to study the effects of implicit assumptions (through introduction of false cues) on problem solving and creative idea generation.

In one study, Smith [3] presented a number of word completion problems to subjects. Some problems were accompanied with positive primes; others were accompanied with negative (blocking) primes. A final group of problems were accompanied with completely unrelated primes. Table 1 shows an example with negative, positive, and unrelated primes given.

TABLE 1

	Type of prime		
Fragment	Blocker	Solution	Unrelated
L_D_ER	LEADER	LADDER	MEMBER
AN_TO_Y	ANCHOVY	ANATOMY	SHERIFF
IN_RT__	INVERTED	INERTIA	UNIVERSE
LE_T_RE	LETTER	LECTURE	TEQUILA

(Source: [3])

Those subjects presented with unrelated primes had a 69% success rate; those accompanied with some positive primes had a 83% success rate; those accompanied with negative primes had the poorest performance: 53% success rate.

2.4 Fixation in Creative Idea Generation

Negative priming can have a negative effect within creative problem solving. In one set of experiments, researchers [4] were looking for negative priming effects in creative idea generation. In one experiment, subjects were asked to generate as many toys as possible within a constrained time frame (20 minutes). Half the subjects were presented with example toys at the beginning of the trial. In all experiments, those subjects who were presented with example toys were far more likely than the control group to generate toy ideas containing features of the examples, even when explicitly instructed not to do so.

Similar experiments given to professional and student engineers also found that generated ideas conformed to examples, when examples were presented in advance [4].

The consistency of the results drawn from these experiments strongly suggest that creative thinking can be strongly inhibited through negative priming, a situation not unlike that brought on by implicit fixation.

2.5 Incubation and Contextualization

The experimental research, only briefly presented here, suggests that probes are highly context sensitive: that is environmental and presentational factors can effect the contents and direction of a probe. If a problem appears to be insoluble, switching the contextual frame in which the solution space is formulated and traversed will oftentimes yield greater success.

This observation will strike the reader as obvious: how many of us, when faced with a difficult problem or immovable creative block, have found better success by simply rephrasing the problem formulation, or restructuring how we are thinking about things or doing things?

3 Computer as ‘Interface’

The computer can be understood as a powerful assistant in recontextualizing our working environment, when conceived as a tool for constructing *interfaces*. In being able to construct interfaces, we become empowered to design the presentational (and *re*presentational) framework according to which we can think about problems, come up with ideas, and execute plans.

3.1 The Task Environment

The notion that context can play a significant role in generating creative ideas and solving problems admits to the importance of the *task environment*. A task environment links the physical environment in which tasks are carried out with a set of internalized conceptual frames determining how one is to go about executing those tasks, settings goals, and accomplishing those goals [5]. Task environments encapsulate historically determined and cognitively acquired know-how with respect to particular problem domains and, as such, determine how plans (and their associated probes) are constructed and deployed.

Over the past 3 decades, the role of the computer has shifted from a batch mode computing vehicle to a virtual environment for accomplishing complex tasks. Computers have become interfaces engendering specific task environments reflecting particular problem and activity domains. As such, when we talk about computers, it is not at all uncommon to speak exclusively about their *interfaces*.

3.2 Cognitive Engineering

At one level, an interface determines what the computer user needs to know in order to effectively execute tasks and accomplish goals. At another level, an interface *orients* cognitive activity, by determining a range of acceptable actions according to an accumulated history of human behavior and activity.

According to interface design theorists like Donald Norman, interfaces work best when they leverage a user’s history of experiences, both cultural and personal, in order diminish the conscious attention required within an interaction. Doorways, telephones, stovetop ranges, and computers should all admit easy use without requiring the user to stop and think about the actual actions required to accomplish desired goals [6]. The approach to interface design these researchers promote is referred to as “user-centered”[7].

Underlying the principles behind user-centered interface design is the idea that between two participants in an interaction (“user” and “system” as the two are often termed) there are a number of gulfs separating the goals and knowledge encapsulated within one system (the “user”) and the presentation of available services and resources of another system (the “system”). The purpose of an interface is to bridge these gulfs in order to make it easier for the “user” to interact with the “system.”

The best interfaces, according to user-centered interface design, are those that give the user a feeling of direct engagement with the objects and processes that are relevant to her/his goals and activities [8]. Rather than having to think in terms of the system s/he is using, the user can remain focused on domain-related concepts, thus freeing attention for domain-centered activities.

Interfaces that follow this approach can embrace a more complicated task environment since the user doesn’t have to remember every little detail relevant only to getting the system to what the user wants it to do.

3.3 “User-centered” Interfaces and *Einstellung*

Under the user-centered design paradigm, the interface is built in order to bring the system in closer conformity to the representational framework familiar to the user *vis* a particular problem or activity domain. One problem with such an approach, however, is that it tends to lock in certain patterns of activity and cognition. Luchins and Luchins [9] did extensive research in the negative effects of habituated behavior, and the *einstellung* (“mental set”) that frequently underlies such behavior. They found that the ability of humans to creatively solve problems and construct ideas are impinged in a task environment that induces habitual behavior. In the language of cognitive psychology, we might say that in an environment that enforces habitual modes of activity, planning, and thinking, a human finds himself stuck with implicit assumptions and trapped by negative primes—oftentimes without even realizing it.

User-centered task environments may be appropriate for tasks like driving a car, using a stovetop range, operating a piece of dangerous machinery, or producing an article. However, they are not always the most appropriate kind of environment for doing creative work.

3.4 Alternatives to “User-centered” Design

In an effort to design interfaces that avoid the effects of *einstellung*, one might bridge the gulfs between user and system by inducing users to adjust their own conceptualization of the problem domain *by forcing the bridging of gulfs upon the user* [8]. In other words, rather than forcing the presentation of the system to conform to the conceptual space of an intended user, why not force an alteration in the conceptual space of the user in order to conform to the presentation of the system?

Such an approach allows users to restructure the way they think about the problems and processes constituting a given domain. By “forcing” users to restructure their thinking, you are enabling them to build new tracks for internal probes and cues.

Such a notion of interface design still constitutes *design*: we are not talking about arbitrary design decisions engineered merely to thwart habitual behavior (though, oftentimes that is enough to unclog the thought process!). If the interface reflects careful thinking and planning, the investment of effort is balanced by the advantage of having gained a fresh conceptual model of an otherwise familiar domain.

Ultimately, such an approach to design constitutes not merely the design of an interface, per se, but, in fact, *the design of the very experiences that humans may have while engaging in creative work.* As design theorist Liam Bannon writes,

> The need is not simply for more detailed psychological models of how people think and communicate, although such models are of course fundamental to the building of more usable systems, but for a more comprehensive, more enlightened view of people that recognizes their need for variety and challenge in the tasks that they perform [10].

3.5 Toward *Paralogical* Interface Design

Postmodernist theorist J.F. Lyotard uses the term *paralogical* to describe a “pragmatics of science” that explicates the “metaprescriptives” of science (i.e. the ‘presuppositions’ underlying scientific practice), and that petitions the players in a given field of activity to develop alternate metaprescriptives [11]. Paralogical activity is intended to render those underlying presuppositions which determine how we think and act within a particular domain of activity—whose presence we no longer even notice—visible and obvious in such a way that we are empowered to construct new ones.

This is tantamount to finding a way, through recontextualization, to transform *implicit* assumptions, whose presence is impossible to cognize, into *intentional* ones, so that we might then better evaluate their use and discard them once they are no longer useful or appropriate to given task or activity.

The notion of *paralogical* interfaces has a precedent in the arts. Artist Robert Morris observed that producing art involves a behavior "aimed at testing the limits and possibilities involved in that particular interaction between one's actions and the materials of the environment" [12]. Jackson Pollock's method of painting exemplifies this principle: laying the canvas flat on the floor, Pollock would stand over the canvas, pouring, dripping, and flinging paint upon its surface. Using this method (which Pollock perfected over a number of years), the artist had to learn the effects of gravity upon paints of different viscosities, and upon the various forms of applications, such as dropping with a brush, pouring from a bucket, and so on. In doing so, he "sought to alter the configuration of problems as they presented themselves for himself as a painter—to restructure the mechanisms, and thereby the descriptive framework according to which painting process might be exercised" [13].

In a similar fashion, composers have fashioned various approaches to 'pre-compositional' activity in their effort to generate musical material and compositional processes. One need only examine the sketchbooks of composers from Beethoven, to Mahler, to Stravinsky, to Elliot Carter, to observe that much of the effort involved in music composition has to do with constructing interfaces and frameworks for developing musical materials and processes for generating musical structures.

3.6 Human/Computer Interaction

Having the ability to design interfaces, through use of the computer, gives us another avenue toward accomplishing this. By understanding human/ computer interface design *paralogically*, we begin to articulate a framework for alternative approaches to interface design—ones that might be more appropriate for certain task domains than user-centered design. Moreover, we develop a conceptual and analytical language for analyzing the many computer systems whose notion of their relevant task domains have transformed how actors within those domains think and act with respect to them.

4 Composition with Computers

For many composers since the late 1950s, the computer has become an important tool. As musicologist Otto Laske has observed, for many composers, the computer forces them "to focus on the pro-active, rather than the re-active, aspect of their activity, [giving] them a chance to choose, rather than suffer, their process" [14].

Laske differentiated *rule-based* composition and *example-based* composition. With rule-based composition, composers explicitly design the rules and procedures according to which musical materials and structures are to be generated and endowed with musical meaning. By contrast, with example-based composition, composers construct materials and musical processes from past "examples" obtained through experience and practice [5].

In truth, rule-based approaches to composition frequently are based, at least in part, on previous models or "examples" of music, even if it is music which composers themselves composed [14]. However, what distinguishes rule-based approaches is that there is an effort to represent otherwise internal processes externally; to *objectify* them so that, as observable objects and processes, they may be consciously molded and manipulated [15].

4.1 Examples

I now present several computer music systems that enable rule-based approaches to music composition in order to exemplify some of the points made thus far.

The program *Project 1* was developed by G. M. Koenig in 1963 and has been used by him and other composers since that time. Using *Project 1*, a composer stipulates, as program input, a set of "structure formulae" whose stochastic structural characteristics range from *order* to *disorder* [16][17][18]. These structure formulae are applied to a repertory of parametric materials (such as pitch, duration, instrumentation, etc). The program generates a list of events, each event defining a potential musical occurance. The task of the composer is to interpret and analyze this list in order to come to some understanding of its structural potential. The composer then begins to design a composition based on her/his understanding of the output.

As Laske perceptively observes, the process induced by a program like *Project 1*, "exposes the transition from analysis to design" [19]. It presents the composer with data that, taken together as well as in its details, presents an "interface" that is at first unfamiliar and foreign. In working with the data, however, the composer learns to integrate her analytical observations of the output with the ideas and plans that led to the initial input in an evolving design model, leading at times to reformulating the input and at other times to reformulating the analysis of the data. The notion of musical form is *emergent*; immanent in the particularity of the composer's interpretive activity.

In Koenig's *SSP*, a software system for composition and sound synthesis, a rule-based approach is again taken [20]. With *SSP*, the composer essentially describes a composition "as one single sound, the perception of which is represented as a function of amplitude distribution in time as sound and silence, soft and loud, high and low, rough and smooth" [20]. The input constitutes a set of data formulations by which everything from individual sounds to aggregations of sounds to entire compositions are fashioned. The output is interpretable as a sequence of integers for digital-to-analog conversion.

With such a system, the methods by which sounds are generated are quite different from those found in more standard sound synthesis practice, in which traditional notions of pitch, timbre, duration and loudness are preserved. Moreover, there is linkage between the generation of individual sounds, and the musical contexts in which those sounds occur that allows the composer to think differently about the relation of musical "microstructure" (i.e. sounds) and musical "macrostructure" (musical patterns and their structures).

More recently, Arun Chandra has developed a method of synthesis, reminiscent of one developed by Herbert Brün (his *Sawdust* system [21]), in which a waveform is not merely a data element, but an object upon which operations can by applied [22]. The *state* of the waveform constitutes a tiny "piece" of that waveform (1 to 2 milliseconds in duration) that is defined by: (1) the number of segments; (2) the "type" of each segment; (3) the sequence of segments; and (4) the number of iterations of the state. A *segment* is a sequence of samples having some discretely defined behavior, or *type* (all the same, moving in a particular direction, etc.). A sequence of segments, and the number of iterations of that sequence define the state of the waveform and determine the acoustical behavior of the resulting sounds.

For each iteration of the waveform, the results of an algorithm are applied to selected segments within that waveform. Changes in segments yield changes in the acoustical structure in the resulting waveform. Each segment is given maxima and minima which determine maximum and minimum growth of the length of the segment, as well as increment values, which determine the amount of change per iteration of the waveform.

As is the case with *SSP*, the composer must synthesize an emergent comprehension of the behavior of the system with a similarly emergent model of musical material: a synthesis which allows the composer to find descriptions and criteria that arise, once again, from the particularity of her activity, rather than from a fixed historical model.

Agostino Di Scipio has pointed out the computer's capacity to assist composers in challenging the dualistic paradigm according to which musical *material* and musical *form* are traditionally separated [23]. Di Scipio posits a theory of sonological emergence in which a composer "imagines and explores possible links between the patterning of atomic details—a ground level process (glp)—and the sound forms which emerge from them—a meta-level process (mlp)" [23]. Di Scipio and Prignano [24] developed an approach to granular synthesis called *functional iterative synthesis* that involves iterated application of difference equations in the generation of sonic material. Such applications have the interesting property of propagating acoustical behavior upwards from the lowest level of sound to higher level patterns of sounds and sound environments [23].

Of particular interest in the approach taken here is that the algorithm not be treated as just another "unit generator" for generating discrete and disconnected sound events (as is the case with more traditional approaches to sound synthesis and the use of MIDI-based synthesizers), but that it be treated as a holistic approach within a more general model of acoustical design (both music *and* sound) [23].

Of related interest, Eckel and González-Arroyo developed a system for composition of music using synthesized sound (called *foo*). Of particular interest is their notion of "sound concept" which "describes the relationship between a musical event and its sound manifestation" in an attempt to allow music representations that are at once locally conceived (for a particular event in time) and globally activated [25]. A sound concept can be understood as "dynamic compound structure, where behavioral laws and signal processing configurations combine to define an object capable of being viewed both as a sound producing entity and as a logical object musically meaningful" [25].

4.2 Discussion

Projects such as these challenge standard notions of the musical and compositional task environment, projecting, instead, a *radical* (that is, from the ground up) approach to musical and acoustical design. Such projects focus on the design of highly particularized task environments in which the problem and task domain is imbued with an *idiosyncratic* interpretation of the possible activities to be taken, as well as the possible artifacts to be produced. Each system defines models with a high degree of systemic integrity: while any one of these systems pose an initial challenge to the composer, the composer is rewarded, after some initial work and experimentation, with a principled presentation regarding possible musical representations. Each forms a novel view of musical and compositional procedure while nevertheless presenting the user with an inarguably “musical” problem and task domain.

5 Conclusion

The preceding discussion attempted to provide an overview of the opportunities presented through the introduction of the computer into the process of designing and producing novel musical structures. Of singular significance is the capacity of the computer in the construction of task environments whose structure empowers the execution of highly demanding creative tasks. In its capacity for the production of *interfaces*, the computer allows the construction of extremely elaborate and rich contextual frameworks in which problem formulation and creative problem solving are abundantly encouraged. Such use of the computer is particularly well understood within the field of music composition, where composers have sought novel means for articulating possible music structures by constructing task environments within which those structures might be envisioned and realized.

References:

[1] Feenberg, A., Subversive Rationalization: Technology, Power, and Democracy, in *Technology and the Politics of Knowledge*, ed. A Feenberg and A. Hannay, Indiana University Press, 1995

[2] Smith, S. M., Fixation, Incubation, and Insight in Memory and Creative Thinking, in *The Creative Cognitive Approach*, ed. S. M. Smith, T. B. Ward, and R. A. Finke, Publishing House, 1995

[3] Smith, S. M., Fixation in Memory and Problem Solving, presented at the Weiskrantz Symposium on Memory, Baylor University, Waco, TX, 1993.

[4] Jansson, D. G., Smith, S. M., Design Fixation, *Design Studies*, 12(1), 1991, pp. 3-11.

[5] Laske, O. E., *Music, Memory, and Thought: Explorations in Cognitive Musicology*, University Microfilms International, 1977

[6] Norman D. A., *Psychology of Everyday Things*, Basic Books, 1988

[7] Norman, D. A., Cognitive Engineering, in *User Centered System Design: New Perspectives on Human-Computer Interaction*, ed. D. A. Norman and S. W. Draper, Lawrence Erlbaum Associates, 1986

[8] Hutchins, E.L., Hollan, J. D., Norman, D. A., Direct Manipulation Interfaces, in *User Centered System Design: New Perspectives on Human-Computer Interaction*, ed. D. A. Norman and S. W. Draper, Lawrence Erlbaum Associates, 1986

[9] Luchins, A. S., Luchins, E. H., New Experimental Attempts at Preventing Mechanization in Problem Solving, *Journal of General Psychology*, 42, 1950, pp. 279-297.

[10] Bannon, L. J., Issues in Design: Some Notes, in *User Centered System Design: New Perspectives on Human-Computer Interaction*, ed. D. A. Norman and S. W. Draper, Lawrence Erlbaum Associates, 1986

[11] Lyotard, J. F., *The Postmodern Condition: A Report on Knowledge*, transl. G. Bennington and B. Massumi, University of Minnesota Press

[12] Morris R., Some Notes on the Phenomenology of Making: The Search for the Motivated, *Artforum*, 8, 1970, pp. 62-66.

[13] Hamman M., From Symbol to Semiology: Representation, Signification, and the Composition of Music Interaction, *Journal of New Music Theory*, 28(2), 1999, pp. 90-104.

[14] Laske, O. E., Toward an Epistemology of Composition, *Interface*, 20, 1991, pp. 235-269.

[15] Hamman, M., Structure as Performance: Cognitive Musicology and the Objectification of Compositional Procedure, in *Otto Laske: Navigating New Musical Horizons*, ed. J. Tabor, Greenwood Press, 1999

[16] Koenig G. M., Project 1, *Electronic Music Reports*, 2, pp. 32-46.

[17] Koenig, G. M., Working with ‘Project 1’: My Experiences with Computer Composition, *Interface*, 20, 1991, pp. 175-180.

[18] Laske, O. E., Composition Theory in Koenig's Project One and Project Two, *Computer Music Journal*, 5(4), 1981, pp. 54-65.

[19] Laske, O. E., Composition Theory: An Enrichment of Music Theory, *Interface*, 18, pp. 45-49.

[20] Berg, P., Rowe, R., and Theriault, D., SSP and Sound Description, *Computer Music Journal*, 4(3), pp. 25-35.

[21] Blum, T., Herbert Brun: Project Sawdust, *Computer Music Journal*, 3(1), pp. 6-7.

[22] Chandra, A., http://grace.evergreen.edu/~arunc.

[23] Di Scipio, A., Formal Processes of Timbre Composition: Challenging the Dualistic Paradigm of Computer Music, *International Computer Music Proceedings*, 1994, pp. 202-208.

[24] Di Scipio, A. and Prignano, I., Sintesi per Iterazioni Funzionali, *LMS Technical Report*, 1994.

[25] Eckel, G. and González-Arroyo, R., Musically Salient Control Abstractions for Sound Synthesis, *International Computer Music Proceedings*, 1994, pp. 256-259.