

A Microworld Approach to the Formalization of Musical Knowledge

Henkjan Honing

*Department of Computational Linguistics, University of Amsterdam, Amsterdam, The Netherlands.
e-mail: henkjan@alf.let.uva.nl*

Abstract: This paper is about the importance of applying computational modeling and artificial intelligence techniques to music cognition and computer music research. The construction of microworlds as a methodology plays a key role in the different stages of this research. Several uses of microworlds are described. Microworlds have been criticized in the domains of artificial intelligence and the cognitive sciences, but this critique has to be seen in its proper context (i.e. in modeling of human intelligence, not as a methodology). It is shown that the microworld approach is still an important methodology in music cognition and computer music research, and a promising strategy in the design of a general representation formalism of musical knowledge.

Key Words: Artificial Intelligence and Music, Microworlds, Knowledge Representation, Music Cognition, Representation of Time and Temporal Structure.

Introduction

Music representation is a research topic with great relevance and application in the fields of music analysis and production (for instance, music notation and retrieval systems), and computational modeling of music perception and performance (for instance, describing mental representations of music). While each domain tends to develop its own specialized representations, the central issue in music representation research is to describe what is shared among these diverse representa-

tions. For example, what makes a chord a chord, and what properties can be generalized over the different representations? This task of constructing a general representation of music is difficult to imagine and to plan, especially since comparable projects of comparable complexity (for example, natural language understanding) have not as yet reached high levels of success. We still lack a general theory of representation, "a sobering fact since our systems rest on it so fundamentally" (Smith, 1991). General representation languages are under development, and there are, besides lots of technical difficulties, still theoretical and philosophical problems of enormous proportions. I nevertheless think that it is very important to look for generalizations and abstractions in the design of a representation of musical knowledge. Since the construction of a complete and general representation of music is still far ahead of us (if not fundamentally impossible), gaining understanding of what can and what cannot be represented, using certain types of formal representations, is far more important and realistic (Honing, 1993). The methodology of constructing microworlds or micro-version programs has turned out to be a successful strategy in building these formalized components of a representational system for music, components that are well understood and generalized in such a way that maintenance and extension are guaranteed.

Microworlds

Many of the microworld ideas stem from the group of researchers that worked at MIT in the seventies (for instance, Abelson, Minsky, Papert,

Henkjan Honing is research fellow at the University of Amsterdam, where he is doing research on the formalization of musical knowledge concentrating on time and temporal structure. With Peter Desain he wrote the book Music, Mind and Machine: Studies in Computer Music, Music Cognition and Artificial Intelligence.

Sussman, Winograd). The notion of a microworld has been described as:

Each model – or “micro-world” as we shall call it – is very schematic; it talks about a fairyland in which things are so simplified that almost every statement about them would be literally false if asserted about in the real world. [. . .] Nevertheless, we feel that they [the micro-worlds] are so important that we are assigning a large portion of our effort toward developing a collection of these micro-worlds and finding how to use the suggestive and predictive powers of the models without being overcome by their incompatibility with literal truth. (Internal MIT memo, Minsky and Papert, 1970; quoted in Dreyfus, 1981)

Papert and his colleagues developed several microworlds for use in an educational context, inspired by the cognitive development theory of Jean Piaget. These microworlds were designed to facilitate learning, and some involved a new programming language called Logo (based on Lisp) embodying the educational philosophy of “learning without being taught” (Papert, 1980). The most prominent example of these microworlds is the turtle-world, which models a world of turtle-geometry (Abelson and diSessa, 1980). Children learned about this world by giving commands to a turtle robot or a turtle image on a computer screen, and building procedures from them. They gained knowledge and understanding of (turtle) geometry simply by exploring the possibilities of this object.

Another often referred to example of the microworld notion is Winograd’s block-world for natural language understanding (Winograd, 1972). Here, by contrast, not the domain (i.e. knowledge about blocks and the ways they can be stacked) but natural language understanding is central; the microworld of blocks merely serves as a toy problem to test the possibilities of a certain approach to natural language processing. This kind of approach, and the optimistic belief that microworlds could simply be combined and extended into a general knowledge representation, prompted much criticism (see for example Dreyfus, 1981) which gave the notion of microworlds bad press (causing Winograd to take the side of his critics, see Winograd and Flores, 1987). This critique, though, should be placed in the perspective of using microworlds to model human knowledge and intelligence, instead of extending the criti-

cisms to the methodology of building microworlds. The methodology offers the researcher the advantages of what Minsky (1987) called “exploratory programming”, i.e. to avoid having to start with a complete and detailed specification. This exploratory programming became an emerging methodology in the seventies, an alternative to more formal approaches, such as logic. In general, the computational modeling approach developed into an attractive alternative to more data-driven types of research (the experimental sciences). It introduced a line of research in music cognition that concentrated on the computational modeling of the processes in perception and production of music, focusing on possible explanations of these processes. The main advantage is that the theories (in the form of a computer program) are described at a level of concreteness that is “open to direct and immediate test” (Longuet-Higgins, 1973), and allows others to repeat the experiments and examples given, as well as test the theory on related data.

As I will show, there are still strong arguments for the microworld approach that make it a valid and important methodology in the research areas of music cognition and computational or systematic musicology.

Building Microworlds as Methodology

Our own experience with microworlds started some ten years ago as an approach in the design of composition systems for music, influenced by the work of the Logo community (Desain and Honing, 1988). Small, but more or less complete, sets of typical composition techniques were grouped into different microworlds. Each set contained simple program-generators (comparable to the *forward* or *right* primitives in the turtle world) that could be combined into more complex but well-understood procedures to generate music. Building microworlds developed over the years into a methodology that accompanied us in different areas of music and AI research, concentrating on the perception and representation of time and temporal structure in music (Desain and Honing, 1992a).

One of the areas in which we applied this methodology was in research on expressive timing.¹ By studying existing formal models of

expressive timing and reducing them to micro-versions that could be applied to the same musical data, we could evaluate and compare them. Having those models on a manageable scale, sharing the same data abstraction, and actually listening to the results while changing the model's parameters, we were able to indicate some of their restrictions. By doing a simple listening test, we could easily show that the implicit or explicit use of a particular representation of time (the so-called "Tempo Curve") is an abstraction with little musical or perceptual value (Desain and Honing, 1993). It represents tempo as a continuous function of score time, independent of the musical material (the events or notes carrying the expression), assuming the possibility to perceive timing without events carrying it.

This prompted us to build a microworld in order to express timing in terms of different kinds of musical structure. It took the form of a Calculus (Desain and Honing, 1992b; Honing, 1992) in which one can describe how particular types of expression are linked to particular types and levels of the musical structure, and how they change under, for example, tempo transformations (for instance, how a grace note's duration scales up compared to a "normal" note with a change of tempo, or, how the depth of a rubato is affected by the density of notes). Although still little is known about this specific musical knowledge and its behavior under transformation, the exploratory microworld gives more insight on how timing is related to specific types of structure² and the way it is affected by specific transformations such as a change of tempo.

In the next stage of this research this knowledge could then become part of the representation. This can be thought of as constructing little machines that have concrete parts and their own particular automatic behavior (for instance, a grace note's duration scales differently under tempo transformations than a normal note). Models of expressive timing, or more general models of music perception could then make use of these "machines" as their primitives, and as such abstract from their behavior and concentrate on the other levels of musical structure. The goal of this research is not to make a model of music perception and performance, but to design a language or

representational system that is powerful enough to express the perceptual aspects of it in a general and elegant way. This work is the object of current research (Honing, 1993).

The Advantages of Actually Programming

In fact, programming a certain idea can provide new insights. Programming actually forces us to answer questions we did not think of before or can lead us to see other ways of programming it (for example, choosing a different data abstraction or control mechanism). A microworld, with of its relatively small dimensions, invites us to do things differently because not all the work (as in a larger system) is dependent on the abstractions chosen. Experimenting with the resulting ad hoc formalization or program may bring out further insights, and provide a real understanding, and, in turn, possibly a new formalization and an adjusted theory. In making problems concrete, deciding what is essential and what is not, and changing knowledge and understanding from being implicit (for instance, hidden in the control structure, i.e. the flow of control) to being explicit (for example, as data structures), problems become objects, objects of thought, that facilitate thinking about them – just as the turtle gave children "an object to think with" (Papert, 1980), helping them to understand more about geometry.

Is a Microworld More or Less than a Theory?

A computational version of a theory in the form of a microworld has a number of advantages. After the formalization of the essence of a theory we can recapture its implications and in the process better understand how to achieve abstractions and true generalizations. To build and, even more important, to use such a microworld formalization will most likely bring out aspects not foreseen during the design of a theory. It makes the theory concrete and verifiable. The construction process itself may even influence the design by revealing flaws and missing aspects (like in the example on Tempo Curves described above). As such, a microworld is more than a theory.

But there are also some negative aspects that can be associated with the construction of programs or microworlds. One frequently sees, in a computational approach to music, that a class

of problems is described (for example, harmonic analysis) followed by a description of a program and a description of the results obtained from sample problems (for instance, certain chord progressions). Often this is just one of a small set of problems with an unclear relation to the class of problems the program or the methods embody. This is what McCarthy (1990) calls the “look, ma, no hands” syndrome: the system works, but little understanding of the domain and problem is gained.³ It is unclear what the program’s limitations are, which aspects are generalizations, which aspects are specific to a particular problem, and which can be attributed to a whole class. If these limitations are not stated along with the program, the program is more or less a black box (although we can look into it, the only thing we see are details), we cannot derive any abstractions from its parts or workings. The program works, but we do not know precisely why and, even more important, we have no idea when it would not work. There is a danger of losing flexibility and awareness of a certain set of untreated problems. As such, a microworld is far from a theory.

Difference from an Expert System Approach

The microworld approach as a methodology might emerge even more clearly when we compare it to the expert system approach. A microworld is not just a small expert system, but is actually its antithesis. Expert systems or rule-based systems accumulate knowledge in the form of a relatively large collection of rules. They describe explicitly what to do in a large collection of specific cases. Extra rules are added to model certain interactions or take care of unwanted interactions (this approach is often negatively called patchwork rationalism). Rule-based systems can be effective when applied to a very restricted domain containing a relatively small collection of rules (Winograd, 1990). As such, they are the opposite of microworlds, in the sense that they are supposed to embody the near-completeness of knowledge with respect to a certain domain (obtained by an over-specification of rules). A rule-based system is capable of reasoning in cases where human beings cannot oversee the consequences any more (as is the case with, for example, nuclear power, law, or medical expert

systems). Microworlds, on the contrary, are designed such that human beings *can* oversee the consequences. Furthermore, where the exploratory microworld serves to *find* all possibilities and consequences of a certain micro-theory, which interactions are important and which could be ignored, a rule-based system is supposed to *describe* all these possibilities and consequences, though they are contained implicitly in a large collection of rules and their interactions. It is therefore peculiar that there is still a body of research that has full faith in this rule-based approach to the modeling of musical knowledge. For instance, the work of Sundberg (Friberg, Frydén, Bodin and Sundberg, 1991) describes a relatively large collection of rules for expression (that can be applied to a score, generating an expressive performance) without worrying too much about their interactions.⁴ Even in the domain of Artificial Intelligence there is some deeply rooted optimism with regard to the possibility of representing the sum total of human knowledge using the existing expert system technology (Lenat and Feigenbaum, 1991; see also critique by Smith, 1991).

The Modularization of Musical Knowledge

The idea that in music “everything has to do with everything”, and the consequent belief in the impossibility of describing aspects of it in isolation, finds a lot of support in ethno-musicological research. Indeed, there is a large amount of knowledge, skills and experiences involved in the description of all aspects of music (dependent on the specific approach to music, be it socially, perceptually, or historically motivated). But the idea to combine all these different perspectives and types of knowledge into one representation is quite useless: a universal representation of music is impossible.⁵ It is therefore preferable to use the term *general*. The definition of “general” is important here, since it makes significant restrictions. By “general” I mean firstly, a representation that describes the measurable and perceptual aspects of music (for example, an acoustic signal) and secondly, the cognitive aspects that are directly involved with this perception (for example, high-level musical notions like metre or tempo). The latter has its problems. The term

“cognitive” refers to models or systems that contain knowledge and process knowledge. But are there any limits on the knowledge we need for our general representational system? Do we have to incorporate – besides, for example, knowledge on metre – information on the mechanics of musical instruments, how the human muscle system works (playing these instruments), what a concert hall is, etc.? We have to be able to restrict the required knowledge.

Restrictions on Modularization

Is it possible to make musical knowledge modular, divide it up in sensible parts? When everything is important, all world knowledge and skills are relevant to such a representational system, and modularization of it will be difficult. We could start with small portions of that knowledge and try to combine them towards a more general knowledge representation. But when knowledge is encapsulated (separated from the rest of the world knowledge), it is difficult, and most of the time even impossible, to determine what knowledge is affected and what knowledge is unaffected by a certain change or addition of new fragments of knowledge to a knowledge base. (This is called the “frame problem.” See McCarthy and Hayes, 1981.) If we think of a microworld as a small knowledge base, the possibility of extending and combining microworlds can be questioned. As an example, when we build a microworld with knowledge on how timing is linked to the rhythmical and metrical structure, how is it affected when we add knowledge on melodic and harmonic structure?⁶

One might then conclude that approaching the problem from the other end is a better path to take: i.e. to start with describing “everything”, the whole of world knowledge and skills. Fodor (1983) takes an important stand on this. He doubts that it is possible to formalize cognitive processes at all. They are part of one central system that is global, non-modular, and therefore cannot – with our current theoretical tools and methods – be comprehended, and cannot therefore be formalized. He considers this lack of understanding as the basis of a failure in formalizing cognitive processes. He thinks the cognitive sciences can be and are successful in formalizing the modular

parts of the mind: the input systems that are “cognitively impenetrable”, like the five senses and language (Fodor, 1983). Only these are a successful domain for AI and psychological research.

The problem now becomes whether music can be considered as being part of this central system, or whether it is a module on its own? It clearly is part of the first if one takes into account all the social and cultural aspects of music; music can be a cognitive faculty among a lot of other things. But, in restricting a representation of music to, first, all the information measurable in the acoustic signal itself and secondly, by the cognitive processes that directly act on it, it seems limited enough to gain some level of success (following Fodor’s argument).⁷ Within this definition I think it is possible to work towards generalizations that can form a basis of cognitive models of important aspects of music.

Conclusion

In retrospect we may generalize that the use of microworlds comes in three versions. First, the exploratory microworld, with which it is easy to experiment with ideas, vague as they are, to gain more insight into the problem to be understood and modeled. Second, the micro-theory microworld, making a reduced version of a theory in the form of a program, so it becomes explicit and allows for tests on completeness and internal consistency. And third, the micro-version microworlds, in which reduced versions of larger programs (or models) are made, preferably sharing the same data abstraction.⁸ In trimming these computational theories down to a bare minimum, they allowed for better and easier comparison, bringing a real understanding of the theory with, more than once, the emergence of more abstract or general notions as a result. But of course, in reality when doing computational modelling all three phases gradually transfer. We may, for example, have a micro-version of a theory for some aspects, but still need more theory on how they relate or whether they are complete. By using this micro-theory version to explore, we might bring further insight which allows us to update and improve the original theory.

This process of reducing problems to their bare essentials, turning them into concrete microworlds

which embody them, does not come for free. The methodology does not help in making the right decisions. A philosophy or strategy has to be there as we step through these phases, to help to decide what is and what is not important. The most important characteristics of a microworld are, besides its exploratory strength, the way it makes abstract problems concrete, the relative ease of finding and making new abstractions and generalizations within and between related microworlds.

Finally, every theory and program will have its limitations. These should be understood and known at all times, and have to be clearly set out alongside the description of the microworld. And, since they only model a very small aspect of the real world, it is important to provide all the information about how to extend and maintain them, allowing other researchers to evaluate the claims made, compare different solutions, and possibly make extensions.

Acknowledgements

An earlier version of this paper benefited greatly from discussions with Peter Desain and remarks by two anonymous referees. Some parts of this text were used in the introduction of Desain and Honing (1992a). The revised version of this paper was written during a two month visit in 1992 at New York University (NYU) supported by the NYU Music Department and a travel grant by the Netherlands Organization for Scientific Research (NWO). The research of Dr Honing has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences (KNAW).

Notes

¹ The study of how deviations in timing in a performance (with respect to, for instance, a metronomical or mechanical performance) are used to communicate structure or the specific interpretation of the performer, and how listeners are able to perceive and appreciate these deviations.

² That they are related we know from a large body of experimental research in Music Psychology on expressive timing. See for example, Clarke (1987), Palmer (1989), or Repp (1992).

³ This is especially true in a connectionist or neurode network approach to music where it is relatively easy to obtain a reasonably working system, without having a real understanding why it works.

⁴ For example, the application of one rule might delay a note a little (with respect to its notated onset in the score) while another rule might alter that same note to be earlier than notated, resulting in no change at all, an interaction not described in any rule.

⁵ The pursuit of it should therefore be rejected.

⁶ Knowledge on melodic and harmonic structure seems to be dependent on what is known or described on time (e.g., rhythm interferes greatly with the perception of melody) – the other way around, starting with time and temporal structure (as we decided in our research), allows one to ignore (for the time being) the melodic and harmonic structure.

⁷ The question, whether music cognition can be described as such a restricted and isolated domain, is still open.

⁸ Making a micro-version of an existing program is an alternative to rational reconstruction, a more established AI evaluation methodology (see for instance, Richie and Hanna, 1990), that encompasses reproducing the essence of an AI program's significant behavior with another program, constructed from descriptions of the important aspects of the original program. In this way, one is able to evaluate the published claims made by those programs.

References

- Abelson, H. and A. diSessa. *Turtle Geometry: Computation as Medium for Exploring Mathematics*. Cambridge, MA: MIT Press, 1980.
- Clarke, E.F. "Levels of Structure in the Organisation of Musical Time". In "Music and Psychology: A Mutual Regard". Ed. S. McAdams. *Contemporary Music Review*, 2, 1 (1987).
- Desain, P. and H. Honing. "LOCO: A Composition Microworld in Logo. *Computer Music Journal*, 12, 3 (1988), 30–42.
- Desain, P. and H. Honing. *Music, Mind and Machine: Studies in Computer Music, Music Cognition and Artificial Intelligence*. Amsterdam: Thesis Publishers, 1992a.
- Desain, P. and H. Honing. "Towards a Calculus for Expressive Timing in Music". *Computers in Music Research*. Vol. 3. 1992b. Also in Desain and Honing, 1992a.
- Desain, P. and H. Honing. "Tempo Curves Considered Harmful". In "Music and Time". Ed. J. Kramer. *Contemporary Music Review* (forthcoming). Also in Desain and Honing, 1992a.
- Dreyfus, H. "From Micro-Worlds to Knowledge Representation: AI at an Impasse". In *Mind Design*. Ed. J. Haugeland. Cambridge, MA: MIT Press, 1981, pp. 161–204.
- Fodor, J. *The Modularity of the Mind: An Essay on Faculty Psychology*. Cambridge, MA: Bradford Books, MIT Press, 1983.
- Friberg, A, L. Frydén, L. Bodin and J. Sundberg. "Performance Rules for Computer-Controlled Contemporary Keyboard Music". *Computer Music Journal*, 15, 2 (1991).
- Honing, H. "Espresso, a Strong and Small Editor for Expression". In *Proceedings of the 1992 International Computer Music Conference*. San Francisco: ICMA, 1992, pp. 215–18.
- Honing, H. "Issues in the Representation of Time and Structure in Music". In *Music and the Cognitive Sciences*. Ed. I. Cross and I. Deliège. *Contemporary Music Review*. London: Harwood Press (1993). Also in Desain and Honing, 1992a.

- Lenat, D. B. and E. A. Feigenbaum. "On the Thresholds of Knowledge". *Artificial Intelligence*, 47 (1991), 185–250.
- Longuet-Higgins, H.C. "Comments of the Lighthill Report". *Artificial Intelligence – A Paper Symposium*. London: Science Research Council, 1973. Reprinted in Longuet-Higgins (1987).
- Longuet-Higgins, H.C. *Mental Processes*. Cambridge, MA: MIT Press, 1987.
- McCarthy, J. M. and P. J. Hayes. "Some Philosophical Problems from the Standpoint of Artificial Intelligence". In *Readings in Artificial Intelligence*. Palo Alto: Tioga Publishing, 1981, pp. 431–50.
- McCarthy, J. "We Need Better Standards for AI Research". In *The Foundations of Artificial Intelligence. A Source Book*. Ed. D. Partridge and Y. Wilks. Cambridge: Cambridge University Press, 1990, pp. 282–85.
- Minsky, M. "Form and Content in Computer Science". In *ACM Turing Award Lectures*. Ed. R. L. Ashenurst and S. Graham. Reading, MA: Addison-Wesley, 1987.
- Palmer, C. "Mapping Musical Thought to Musical Performance". *Journal of Experimental Psychology: Human Perception and Performance*, 15 (1989), 331–46.
- Papert, S. *Mindstorms*. New York: Basic Books, 1980.
- Repp, B. H. "Diversity and Commonality in Music Performance: An Analysis of Timing Microstructure in Schumann's "Traumerei". *Journal of the Acoustical Society of America*, 92, 5 (1992), 2546–68.
- Richie, G. D. and F. K. Hanna. "AM: A Case Study in AI Methodology". In *The Foundations of Artificial Intelligence. A Source Book*. Ed. D. Partridge and Y. Wilks. Cambridge: Cambridge University Press, 1990, pp. 247–65.
- Smith, B. C. "The Owl and the Electric Encyclopedia". *Artificial Intelligence*, 47 (1991), 251–88.
- Winograd, T. and F. Flores. *Understanding Computers and Cognition. A New Foundation for Design*. Reading, MA: Addison-Wesley, 1987.
- Winograd, T. *Understanding Natural Language*. New York: Academic Press, 1972.
- Winograd, T. "Thinking Machines: Can there Be? Are We?" In *The Foundations of Artificial Intelligence. A Source Book*. Ed. D. Partridge and Y. Wilks. Cambridge: Cambridge University Press, 1990, pp. 167–89.