Thor Magnusson
School of Arts and Media
University of Brighton
Grand Parade
Brighton BN2 0JY UK
T.Magnusson@brighton.ac.uk

Designing Constraints: Composing and Performing with Digital Musical Systems

Currently, a musician working with digital technology is faced with a panoply of musical tools that can be roughly characterized by a split between ready-made music production software on the one hand, and audio-programming environments such as SuperCollider, CSound, Pure Data, Max/MSP, ChucK, or Audiomulch (to name but a few) on the other. Problems with the former lie in the conceptual and compositional constraints imposed upon users by software tools that clearly define the scope of available musical expressions. It is for this reason that many musicians, determined to fight the fossilization of music into stylistic boxes, often choose to work with programming environments that allow for more extensive experimentation. However, problems here include the practically infinite expressive scope of the environment, sometimes resulting in a creative paralysis or in the frequent symptom of a musician-turned-engineer. Consequently, a common strategy can be detected, defined here as that of *designing constraints*, where the instrument designer, the composer, or the performer (a distinction often irrelevant in these systems; see Drummond 2009) devises a relatively high-level system of constraints, encapsulating a defined space for potential expression, whether of compositional or gestural nature.

This article engages with this situation by exploring the Human–Computer Interaction (HCI) concepts of *affordances*, *constraints*, and *mapping*. We examine how computational systems of musical expression always involve the establishment of a particular stratum that provides certain affordances to the musician, while concurrently posing important constraints. The article studies how constraints can serve as sources for creative explorations. For this purpose, three systems are analyzed: the mLog, Phalanger, and the ixi lang. All these systems are instantiations that proscribe complexity in favor of a clear, explicit space of gestural trajectories and musical scope. Yet, in their nature as digital systems, we find that the named concretizations are always arbitrary, dynamic, and highly transient, posing problems for the instrument's identity and historical continuity.

Computer Music Journal, 34:4, pp. 62–73, Winter 2010

Affordances and Constraints

The phenomenological method of philosophical inquiry, founded by philosophers such as Husserl, Heidegger, and Merleau-Ponty, has influenced much work in HCI (e.g., Winograd and Flores 1986; Dourish 2001). A related, yet narrower, focus of the relationship between the human and the world has been studied under the terms of ecological psychology. The ecological approach to cognition was developed by psychologist James J. Gibson, who studied human perception and the environment as a dynamic system (Gibson 1979). The field of HCI has incorporated many important concepts derived from ecological psychology, such as ecological affordance and constraint. The former is more commonly used within HCI (e.g., Norman 1988; Gaver 1991), but this article argues that in the context of musical interfaces, the latter might be more pertinent.

Affordances

Gibson (1979, p. 127) initially defined an environmental affordance as "what it offers the animal, what it provides or furnishes, either for good or ill." In this definition, affordances are properties of the relationship between the environment and the agent (human or animal). The relationship consists of a mapping between the properties of the environment to the potential actions of the agent. An instrument such as the violin affords certain actions to the human that it does not afford to a bee. Influenced by Gibson, Donald Norman, working in the field of HCI, introduced the idea of *perceived affordances* (Norman 1988): the properties that the agent perceives as possible actions upon an object. This is a narrower definition, as Gibson's original definition of affordances saw them as existing independently of the agent's perception—a view supported by Gaver (1991), who talks about perceptible, hidden, and false affordances. Norman's view has been influential in the field of HCI to the degree that most contemporary designers are aware of the importance of affordances in system design.

Affordances have also been defined as entirely subjective. Vera and Simon (1993) define affordances as "carefully and simply encoded internal representations of complex configurations of external objects, the encodings capturing the functional significance of the object." Costall (1995) extends this definition of affordance to also signify a social construction, something that people learn from each other in every culture. A physical object will thus have different affordances in different cultures. However, to get around the relativism of the social constructivist argument, he proposes the "canonical affordance" (Costall 1997) of an object as derived from its name; one hammers with a hammer, wrenches with a wrench, and thus "computes" with a computer.

Ian Hutchby (2001) disagrees. For him, affordances are important precisely because they are not culture-specific. They can provide objectivity, constituting a common platform that exists beyond all cultural difference. Hutchby claims that—as opposed to the views of Grint and Woolgar (1997), who propose a technological hermeneutics—technology exists in a real, physical world that is not reducible to textual interpretation: "Different technologies possess different affordances, and these affordances constrain the ways that they can possibly be 'written' or 'read'" (Hutchby 2001, p. 447). A musical instrument thus affords certain ways of playing, but at the same time it allows for a cultural reading of its expressive scope: It is clear how the affordances of certain instruments change when they are taken into use in different cultures.

One could arguably take a compromise stance between the objective and cultural definitions. From the perspective of phenomenologically inspired enactivism (Varela, Thompson, and Rosch 1991), it can be argued that it is impossible to talk about affordances that are not grounded in a specific culture. This article therefore rejects the subjective view of affordances maintained by Vera and Simon (1993). Affordances are not representational structures of the object in the subject's mind. They are rather seen as potential applications derived from the agent's embodied relationship with the object in the enactive sense, therefore maintaining the original meaning of affordances as non-representational. However, as this section has illustrated, it is clear that there are certain problems using the term "affordance," considering its highly varied interpretations and definitions.

Constraints

Margaret A. Boden defines constraints as one of the fundamental sources for creativity: "[F]ar from being the antithesis of creativity, constraints on thinking are what make it possible. . . . Constraints map out a territory of structural possibilities which can then be explored, and perhaps transformed to give another one" (Boden 1990, p. 95). For Boden, the continuity of cultural constraints constitutes the possibility to evaluate creative work, or to recognize ideas as creative. All cultures are founded on constraints; they are the rule-sets that maintain dynamic unity. Constraints can be implicitly understood and explicitly formulated. Constraints are also a term used in compositional theory (Courtot 1992; Ebcioğlu 1992; Anders and Miranda 2008). Here, constraints are seen as compositional rules that the computer (or the human) must follow.

Cultural or ideological constraints are not the only constraints. Composers and performers must also engage with the physical constraints of the musical instrument that they are composing for or playing. The physical constraints of the piano define its expressive scope, just as the theoretical constraints of a compositional system (such as the 12-tone system) define the compositional scope. Furthermore, the pianist's style of playing the composition is highly dependent on cultural constraints.

Various models of constraints have been proposed. Norman (1999) brings forth a model of physical, logical, and cultural constraints. *Physical constraints* define what can be performed with the physical environment, such as the mouse cursor's not leaving the computer screen. *Logical constraints* outline how human logic inductively informs us about the environment. *Cultural constraints* are shared by cultural groups that form specific conventions (Norman 1999). In the field of music, Pearce and Wiggins (2002) define the working constraints of a composer into three categories: *stylistic constraints*, where the composer is working within the limits of a specific genre or style; *internal constraints*, the logical possibilities of how the piece can progress according to the rule set that has been implicitly or explicitly set; and *external constraints*, the need to be sure that the piece is physically possible for a performer.

In the context of this article, an alternative model of constraints is presented. It focuses on the philosophical–technological relationship between the human and the tool, and it takes into account the social context in which they exist. Here, I define *subjective constraints* as referring to the expressive limitations that face the thinking, creative, performing human. They originate from a long inscription of the musician into a musical tradition, where the musician habituates (Bourdieu 1990) the practices of the tradition. *Objective constraints* represent the physical limitations of the environment or physical material and the designed constraints of human tools. Examples include the inability to play very high-pitched notes on a double bass or to control the amplitude envelope of a note played on a piano. These constraints also exist in software. *Cultural constraints* are the conditions in which both technology and ideas exist. They affect the musician who becomes constrained by the technology at hand and the cultural values that underpin its design. In this context, it is important to traverse beyond the simple dualism between sociological constructivism and technological determinism. Technology is defined by culture but equally defines it as well (Latour 1994).

Musicians often have problems breaking an instrument's cultural constraints. These problems are partly owing to extensive training in a particular musical culture where the instrument has become the vehicle of certain musical practices. The cultural constraints thus inform subjective constraints, where musicians are defined by the implicit or explicit rules of the musical tradition in which they participate. Objective constraints affect creativity as well. Here, the musician might reject the script of the instrument and reinterpret it (Akrich 1992). An example is the "table-top guitar," where an electric guitar is used in a manner for which it was not designed. For Boden (1990), constraints such as these constitute a search space in which three types of creativity can take place: combinatorial, explorative, and transformational. It is beyond the scope of this article to explore Boden's theory of creativity, but it is important to point out that the study of constraints is a complex field that can be beneficial to designers of musical instruments, composers, and performers alike.

Affordances or Constraints?

In HCI, affordance is typically defined as the perceived capacity of a system for certain actions. It is a feature of the object that can be acted upon: a chair on which to sit, a button to press, a mouthpiece into which to blow. In the context of musical instruments, both acoustic and digital, constraints can be defined as their limitations. As opposed to affordances, the constraints of a musical instrument are often not directly perceptible at the initial encounter. They become understood through engagement and experience. Similarly, the fugue is a musical form that is defined by its constraints—what is allowed and what not. This is different from how the composer considers the affordances of the instruments they compose for, or the affordances of music in the sense of what it "makes possible" (DeNora 2003, p. 46), such as emotional catharsis or dancing (Clarke 2003, p. 120). The two terms are complementary as they focus on different aspects of the same system.

Simple physical objects have easily detectable affordances. The hammer invites hammering; the knob of the mixing console invites turning. Affordance thus appears as an HCI term for early computer systems, where user interaction happens through low-bandwidth input channels, and the tasks to be performed are relatively simple. Contrarily, in complex virtual systems, such as digital musical instruments, affordances can be imperceptible. Thus, the term constraints proves to be more descriptive of what happens in the design and the use of these systems. Instrument makers actively design affordances according their understanding of musical performance and composition. Users, if not the designers themselves, will then engage with the instrument and learn about its expressive potential through experience, building an embodied tacit knowledge (Polanyi 1966) of the system. Owing to the complexities involved in technological artifacts that originate in such an integrated cultural practice as music, learning a digital musical instrument is therefore more appropriately described as "getting a feeling" for the instrument's constraints, rather than engaging with its affordances. If the focus in the design of a door opener is its affordance, the focus in the design of a digital musical instrument goes beyond that to what it can unveil and where its limits lie.

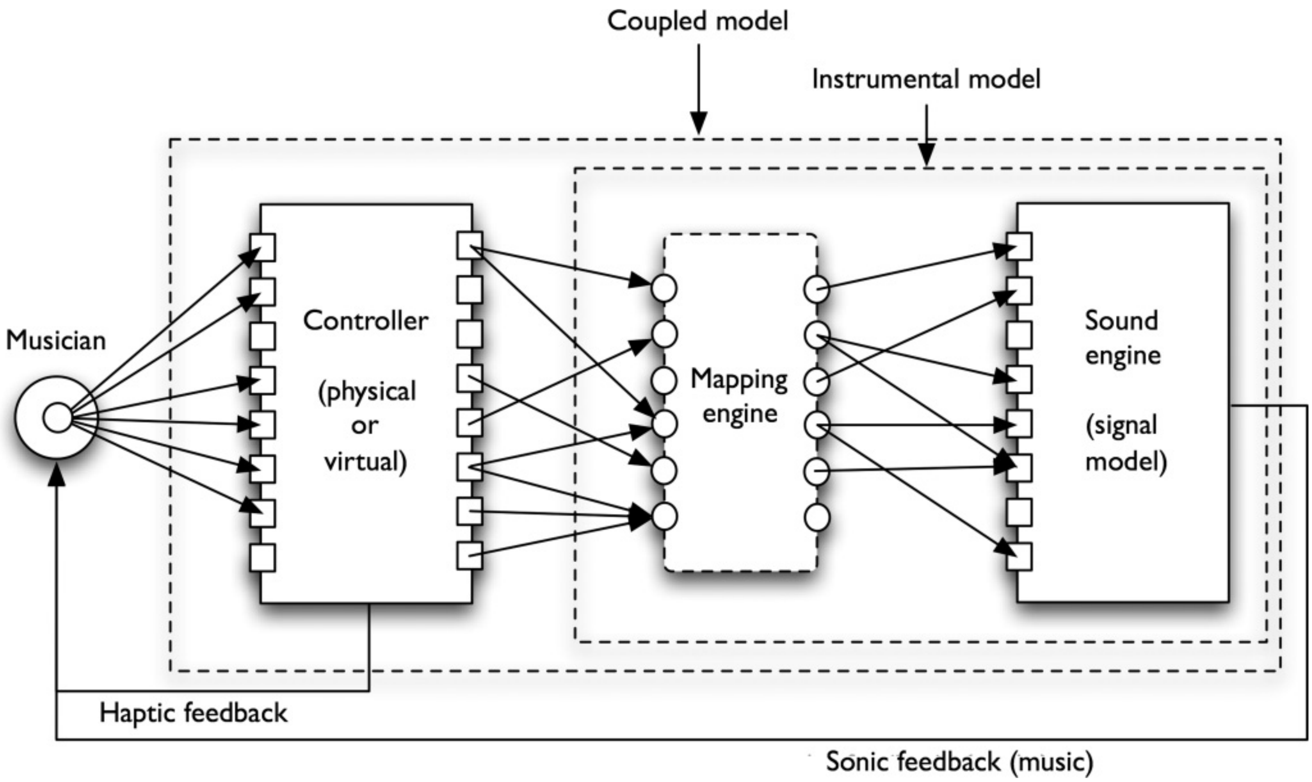
Mapping as Designing Constraints

Mapping is perhaps the most integral feature of new digital musical instruments (Winkler 1995; Chadabe 1997; Bongers 2000; Hunt and Kirk 2000; Orio, Schnell, and Wanderley 2001; Jordà 2005). Its prevalence ranges from ergonomic strategies to compositional features. Some authors (Bongers 2000; Jordà 2005) advocate the view that the interface and the sound engine should be considered holistically as one system, as a unified instrument for expression. Such a view can be beneficial, as the instrument is easier to understand by a composer or a performer, perhaps helping the instrument to gain historical continuity. However, this dream of the digital musical interface to establish itself, to concretize (or *pointillize*, in the lingo of actor-network theory) might be a mere pipe dream. Even if the instrument is presented as a unified object, it will always be characterized by the split between the interface and the sound engine, connected by a mapping engine of diversified complexity. The sound and mapping engines serve as the core of the digital musical instrument; they are its "real body." This is the location where constraints are defined and the instrument's functionality constructed. However, unlike the case of acoustic instruments, such design is rarely final owing to the highly arbitrary nature of digital design, which is more or less free from material constraints. The digital musical instrument is perpetually evolving, although a few designer/performers have made efforts to "freeze" the state of the instrument to gain better mastery of it (Waisvisz 1999).

The design of a musical instrument or a composition is a design decision conditioned by the properties found in the source material. These properties map to the constraints described previously, namely, objective constraints (including the affordances of the physical gestural interface and the limitations of the programming language, protocols, or hardware), cultural constraints (the style of music for which the system is designed), and subjective constraints (the background and experience of the designer). However, even if controllers, programming environments, or musical styles are broader than any one composition or musical instrument, these elements represent a musical system that is necessarily conditioned by an intricate process of analyses, categorizations, normalizations, abstractions, and constructions, where the design paths taken are often determined by highly personal, culturally conditioned, and often arbitrary reasons. Considering all the available parameters and functions, mapping should be defined as a compositional process that engenders a structure of constraints.

As mentioned, the distinctions between the composer, performer, and instrument designer become blurred in progressive digital music. The creator is typically one and the same, which is why Jordà (2005) uses the term "digital luthier" for those who not only build but also perform with their instruments. Pérez, Knapp, and Alcorn (2007) make a distinction between what could be defined as composition-based and instrument-based approaches to the design of digital musical systems, prompting Paine (2009) to ask where one

Figure 1. Typical model of the musical interface (Wanderley 2000; Leman 2008; Wessel and Wright 2001). Here, the split between the gestural interface (either virtual or physical) and the sound engine is bridged by the mapping engine. However, the instrumental model is defined as the core of the instrument, where constraints are programmatically defined.



can define the location of the instrument. Figure 1 attempts to define the instrument's (and equally the composition's) core in the instrumental model. Through this integrated and complex process of design, the results will always be highly personal abstractions that will influence and instigate ideas in the musician using the system. This has provoked the realization in recent work that the system itself becomes an important actor in the ecosystem of the musical performance, not merely influencing and directing through musical encodings, but also extending and augmenting the cognitive and physical capacity of the musician (Gurevich and Treviño 2007; Waters 2007; Bown, Eldridge, and McCormack 2009; Magnusson 2009).

Three Musical Systems That Have High Constraints

All musical systems, from compositional theories to musical instruments, can be seen as systems of constraints. These systems can be located on strata of expressivity. Some are intended to be platforms for general composition or instrument-making, whereas others are abstracted systems for particular purposes. As a case study, I refrain from analyzing the ixiQuarks software, deliberately designed as systems of high constraints (Magnusson 2007). Instead, I will describe my experience as the user of two systems with which I have had the good fortune to compose: mLog by the Owl Project group and Phalanger by Chris Kiefer. I will also describe ixi lang, a live-coding system I wrote in SuperCollider. The creation of these musical systems involved first a thorough exploration and understanding of the source material (hardware, controllers, and programming languages), and second, abstractions or concretizations built on those platforms, equally definable as instruments or compositions.

Figure 2. Typical mLog. The wooden log contains the ingredients that make up a constrained expressive interface. A USB or Bluetooth connection with the computer sends OSC-formatted information from the MUIO chip to the programming language.

mLog

The mLog is a sensor interface designed by the Manchester-based art collective Owl Project. It is a wooden log containing a MUIO sensor interface (www.muio.org) that receives eight digital and four analog sensor inputs. As seen in Figure 2, the mLog has two switches, two potentiometers, six pushbuttons (four on the back), and a two-dimensional accelerometer. It is a perfect example of a physical interface with clear affordances. For the author, the objective constraints of the mLog provided a liberation from complexity and the infinite choices provided by an audio programming language such as SuperCollider. By being given the affordances in the form of a physical entity, the design of the instrument predominantly involved decisions on how to map the interface functionality to a custom sound engine. The composition was therefore highly inspired and constrained by the interface itself. However, as Figure 1 indicates, it should primarily be located at the level of the audio and mapping engines.

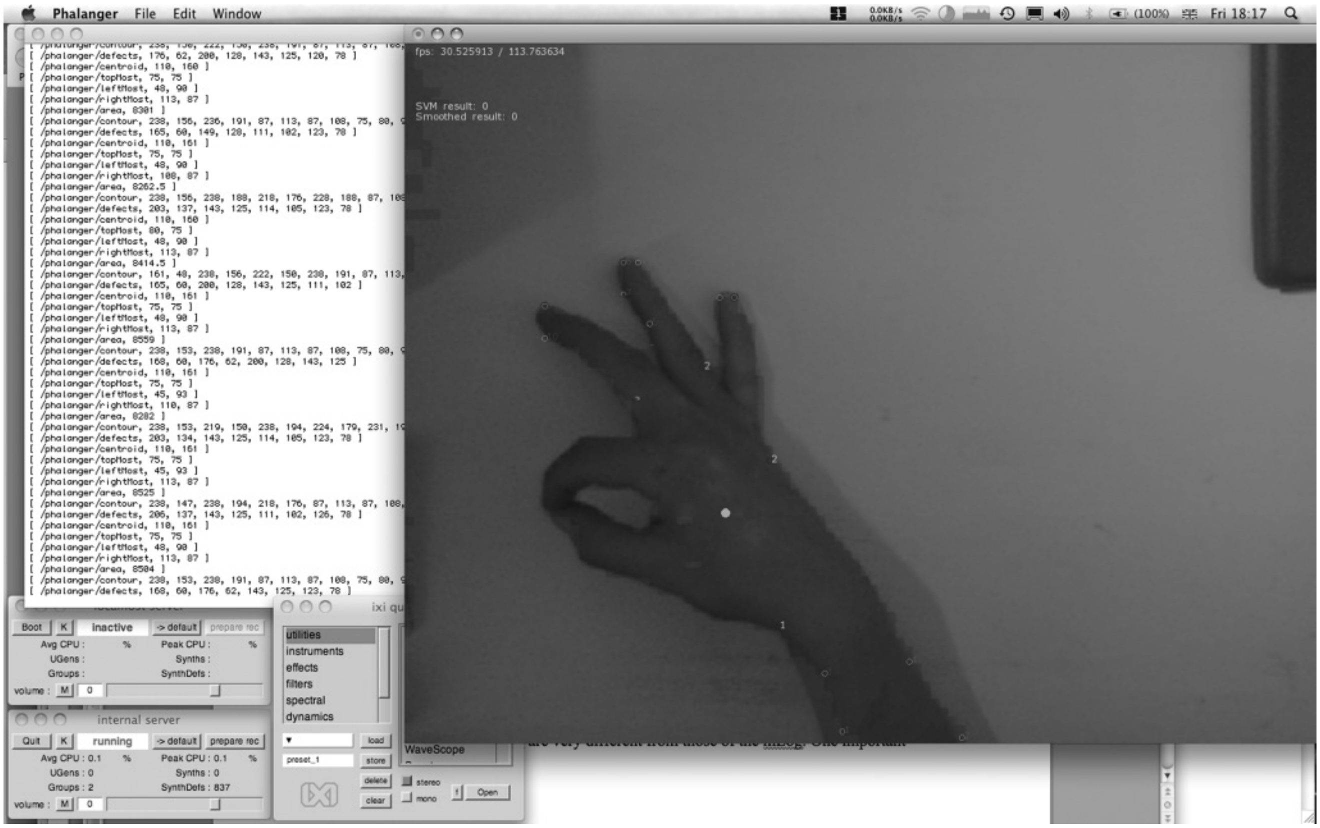
Three compositions were written for the mLog; these represent three different musical instruments. Although the physical interface remains the same, its instrumental functionality differs completely in all three manifestations as a coupled instrument. As its designer, the author felt strongly that the main "body" of the instrument resides at the level of code. The physical interface is merely the control mechanism providing certain affordances for physical action, but mentally, the performer is engaging with attributes and constraints defined in code. Therefore, the physical controller itself, the mLog, could easily be exchanged for a controller with similar affordances (for example a Wiimote or an iPhone) without much change in musical expression. For this reason, I argue that the primary instrumental model (with regard to the question of where the instrument is) should be defined as the mapping engine and sound engine combined, as illustrated in Figure 1. Accordingly, it is the instrumental model that primarily represents the instrument's constraints.

Phalanger

Phalanger is a computer-vision system intended to facilitate musical control (Kiefer, Collins, and Fitzpatrick 2009). It contains a neural network that detects skin color and a support vector machine (Christianini and Shawe-Taylor 2000) that can be taught hand movements defined by the user. This training consists first of detecting skin color and the background, and second of learning hand gestures. Figure 3 shows how the application detects the skin color of the performer and recognizes the gesture performed. The gestures are given their unique integer identification (ID) numbers. After the training process (which takes only a few minutes), the system outputs Open Sound Control (OSC) information containing the normalized centroid, contour points, and, most importantly, the gesture IDs.

The affordances of Phalanger are very different from those of the mLog. There are certain important limitations derived from the nature of the system itself. For example, it only detects one hand, the light must remain constant, and the background must be the same throughout the session. Apart from the system's objective constraints, such as the ability to recognize shapes, other limitations primarily involve the flexibility of the hand and the user's capacity of memorizing hand gestures (which are rarely symbolic except for speakers of

Figure 3. Phalanger in action. The performer forms a hand gesture that has acquired an identity through training of the neural network. Phalanger sends OSC information from the video-analysis application to the programming language.



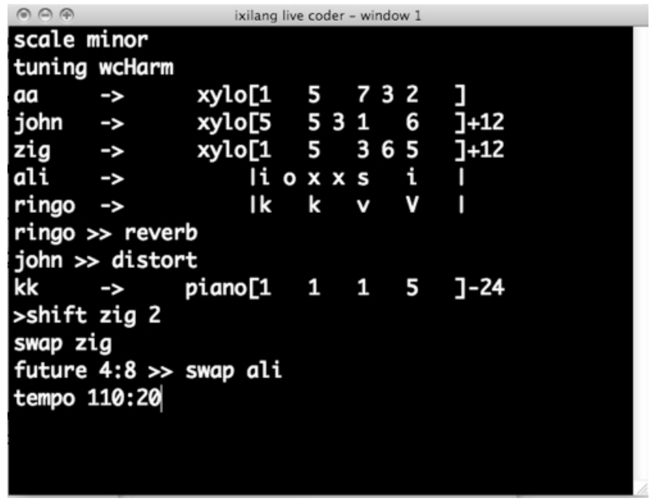
sign languages). For this author, designing musical systems for Phalanger was an intriguing process. A current of gestural data flows from Phalanger into SuperCollider, which is used as the mapping and sound engine. Any gesture ID can be mapped to any musical event, and in the main composition, a decision was taken to use gesture IDs to instantiate a corresponding number of complex SuperCollider synths whose parameters could be controlled with free hand movements (i.e., using centroid and contour data).

Phalanger poses an interesting addition to the model presented in Figure 1. As the neural network can be taught hand gestures and these saved into a gesture file, Phalanger can be used to load in different files of different gesture combinations. Each training session therefore yields a new controller. Different controllers can control the same sound engine, or alternatively, one controller can steer many sound engines. Again, the attempt to define a unified instrument feels unnatural, as the core of the instrument can be found in its mapping. In terms of performer skills, the author was able to build up a repository of hand gestures that were used as symbols for features in the sound system. However, practice is needed for the proper incorporation of these gestures. Continuity with these instruments is unlikely to happen, as every session prompts new ideas, new connections in the mapping engine, or new features in the sound engine. For the author, the instances of musical instruments or compositions built in Phalanger will always be of a highly transient nature.

ixi lang

A typical problem for the live coder is the high level of expertise required for such performance (Nilson

Figure 4. Screenshot of the ixi lang. Each line is written and evaluated in real time, giving the performer full control of the music, but within set constraints inscribed in the programming language itself.



2007). Very few performers are able to exhibit those skills without consistent dedication to practice (Sorensen and Brown 2007). Although fascinated by certain virtuosic live coders, it seemed to this author that such incorporation of dexterity strives against the primary rationale of the mechanical computer, namely, the automation of rote tasks and the augmentation of mental capacity. The decision was therefore taken to design a musical live coding programming language that frees performers from having to think at the level of computer science, allowing them to engage directly with music through a high-level representation of musical patterns. Naturally some degree of algorithmic thinking and consideration of syntax is required.

The ixi lang is a live coding programming language whose interpreter is built in SuperCollider, thus concomitantly gaining access to the underlying power of that environment. However, unlike SuperCollider, the aim with the language was to create expressive constraints. Inspired by operator overloading in C++, the live coding systems of Alex McLean (2004), and esoteric languages such as Whitespace and Brainf _ _ k (see www.esolangs.org), the ixi lang was designed as a high-level system that affords certain types of musical patterns, but excludes others. As such, the system itself becomes a compositional form. Here, constraints inherent in the language are seen as providing freedom from complexity, yet defining a large-enough search space (Boden 1990) for musicians other than the author to explore and express themselves. As seen in Figure 4, the language is very simple and intuitive for the audience. Code can be written that changes other code (and updates the code in the same document), which allows for complex structures and changes over time that are not directly called by the performers.

The ixi lang affords a specific set of musical activities. It provides a scaffold for externalizing musical thinking (Clark 2008) and, through its simplicity, attempts to ease the live coder's cognitive load. As a live-coding system, it goes further than most common live-coding environments in providing a simple, high-level platform for musical expression. As the system is written in SuperCollider (SC), normal SC code can easily be written in the same document, allowing the user to tap into the extensive scope of SuperCollider itself. Learning the affordances of ixi lang as presented in its language constructs might take a couple of hours, but getting an overview of the system's constraints can take many long sessions of practice.

Comparison of the Three Systems

Of the three systems, it is perhaps only the mLog that lends itself to be effectively studied by the theory of affordances. The mLog has perceivable affordances (i.e., the buttons, the knobs, and the hidden accelerometer). Phalanger has no clearly identifiable perceivable affordances. Its physical interface is the camera, but ultimately it is the user who defines the functionality of the system by training its gesture-recognition algorithm. Similarly, the ixi lang has affordances in terms of the methods and functions the system provides.

For this reason, it is argued here that the focus, when designing and analyzing new interfaces for musical expression, should rather be on constraints than affordances, although the latter are always naturally present. It could be roughly stated that affordances have to do with usability, whereas constraints define the limits of musical expression. This can be clarified by the fact that after the initial encounter with a musical instrument (when its affordances are studied), the performer spends more time in exploring and engaging with the instrument's constraints. A recent survey (Magnusson and Hurtado Mendieta 2008) has shown that people enjoy and are inspired by exploring the limitations of digital musical instruments.

Systems like the three described here are so open and flexible as interfaces that it is only when they have been given rigid mapping to sound engines that they gain their function—and indeed identity—as expressive musical systems. This process of creating identity is a process of designing constraints. The performer of the system, when practicing and performing with it, is mentally engaging with the system's constraints rather than affordances. The constraints can be the source of creative inspiration, thus freeing the composer from too much choice and defining the search space by outlining musical rules. This is typical and common in the world of acoustic instruments, but in computer music, these constraints must be actively and consciously defined by the designer/composer.

Virtuosity as Mental Skill in Digital Musical Instruments

The three systems analyzed herein can be defined as being idiosyncratic; they are adapted to personal artistic expression, thus obfuscating the distinction between the instrument and the composition. This fact illustrates how unlikely it is that many of the digital musical systems developed today will establish themselves in the manner of our beloved acoustic instruments. History, legacy, and tradition are not the strong features of new musical interfaces. There are many reasons for the transient nature of the new systems, the most obvious being that they are systems of high-level design of constraints, originating from the more general expressive mold of the programming environments. Code can thus be seen as the wood and strings of the digital instrument maker, the difference being that in building the digital system, a great deal of musical theory is inscribed in the tool, and consequently the performer's agency is prescribed. This happens to some extent in acoustic instruments as well, but to an intensified degree in digital systems.

In the three systems discussed, it is clear that the primary skills demonstrated are not at a level that is directly musical or "instrumental." Rather, we find an expertise involving the knowledge of electronics, computer science, artificial intelligence, and digital audio synthesis. The primary virtuosity is not at the level of the instrument itself or in the relationship between the agent and the object, but rather below the instrument at the strata of hardware and code. Similarly, in generative creative systems, the designer defines the potential semantic space based on rules that effectively outline the structural limits of the possible. Virtuosity in contemporary musical composition can therefore be defined as the skill of designing and understanding constraints.

The digital musical instrument is a system of constraints, both ergonomic and music-theoretical, that is determined primarily by two factors: the musical culture in which the designer is located (cultural and subjective constraints) and the expressive scope of the programming language or the hardware of which it is created (objective constraints). The system becomes an actor (Latour 1994), a container of musical theory in the form of an epistemic tool (Magnusson 2009) that has a performative and mimetic agency as a behavioral object (Bown, Eldridge, and McCormack 2009). In new musical instruments created with general and diverse building blocks, the rationale for creating high-level constraints is primarily to engender an identity, a musical world that is simple, intuitive, and direct. Virtuosity in new digital instruments thus relates to the understanding of the system's core, an understanding typically achieved from the process of being its designer. This type of virtuosity is not a relationship by a performer and the perceived affordances of the instrument (an association found in acoustic instruments), but rather a habituated (Bourdieu 1990) incorporation (Hayles 1999) of the system's constraints achieved through a knowledge of its material, its mapping engine, and the exploration of its expressive limits.

Conclusion

From both informal and formal surveys (e.g., Magnusson and Hurtado Mendieta 2008) of my musical systems, I have realized that in learning the instrument and engaging with its expressive potential, people are only initially concerned with the affordances of the interface. The main bulk of the time spent in learning the instrument involves building a habituated mental model of its constraints. Similarly, the Phalanger affords hand-gesture recognition, but in performance, the user focuses primarily on the relationship between gestures and parameters in the sound engine. These parameters define the instrument more than any physical camera or screen-based representation. The ixi lang provides a world of constrained expression. It fortifies simple musical activity and excludes the bewildering complexity of SuperCollider itself, although one can make use of that language as well.

This article presented three highly diverse musical systems that are abstracted from more complex systems. As abstractions, they become deliberate designs of constraints, where the making of the instrument involves composition, or alternatively, composing involves instrument design. The instrument presents the affordances and some objective constraints, but it is at the level of the sound and mapping engines that we find the field of constraints that becomes the conceptual space (Boden 1990) for the performer or composer to explore. The design of a new interface for musical expression involves the provision of affordances in terms of hardware or software features, but the primary character of the instrument is defined by its constraints. Virtuosity of new musical instruments is therefore not to be found at the level of the interface itself where the performer's body interacts with perceived affordances of the physical interface, but at the level of code or hardware of various strata, where the structure of the search space is defined and limitations are set.

To conclude, it could be said that affordances and constraints in musical instruments are two sides of the same coin, but with a change of focus where affordances point to features that make things possible and constraints define the limits of the possible. Composing an instrument therefore implies some degree of affordance design, but the core activity typically involves the iterative process of experiencing and adopting the system's constraints. Through this process, a limited artifact (the instrument) is abstracted out of another more general artifact (the programming language), and a coherent expressive structure of musical possibilities emerges.

References

Akrich, M. 1992. "The De-Scription of Technical Objects." In W. Bijker and J. Law, eds. *Shaping Technology/Building Society*. Cambridge, Massachusetts: MIT Press, pp. 205–224.

Anders, T., and E. T. Miranda. 2008. "Higher-Order Constraint Applicators for Music Constraint Programming." *Proceedings of the 2008 International Computer Music Conference*. San Francisco, California: International Computer Music Association. Available on-line at http://quod.lib.umich.edu/i/icmc/.

Boden, M. A. 1990. *The Creative Mind: Myths and Mechanisms*. London: Wiedenfield and Nicholson.

Bongers, B. 2000. "Physical Interfaces in the Electronic Arts. Interaction Theory and Interfacing Techniques for Real-Time Performance." In M. Wanderley and M. Battier, eds. *Trends in Gestural Control of Music.* Paris: IRCAM, pp. 41–70.

Bourdieu, P. 1990. *The Logic of Practice.* Cambridge, UK: Polity Press.

Bown, O., A. Eldridge, and J. McCormack. 2009. "Understanding Interaction in Contemporary Digital Music: From Instruments to Behavioural Objects." *Organised Sound* 14(2):188–196.

Chadabe, J. 1997. *Electric Sound: The Past and Promise of Electronic Music*. Upper Saddle River, New Jersey: Prentice Hall.

Clark, A. 2008. *Supersizing the Mind: Embodiment, Action, and Cognitive Extension*. New York: Oxford University Press.

Clarke, E. F. 2003. "Music and Psychology." In M. Clayton, T. Herbert, and R. Middleton, eds. *The Cultural Study of Music: A Critical Introduction.* London: Routledge, pp. 113–123.

Costall, A. 1995. "Socializing Affordances." *Theory and Psychology* 5(4):467–481.

Costall, A. 1997. "The Meaning of Things." *Social Analysis* 41(1):76–86.

Courtot, F. 1992. "Logical Representation and Induction for Computer Assisted Composition." In M. Bablan, K. Ebcioğlu, and O. Laske, eds. *Understanding Music with AI.* Cambridge, Massachusetts, and Menlo Park, California: MIT Press and AAAI Press, pp. 156–181.

Cristianini, N., and J. Shawe-Taylor. 2000. *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods*. Cambridge, UK: Cambridge University Press.

DeNora, T. 2003. *After Adorno: Rethinking Music Sociology*. Cambridge, UK: Cambridge University Press.

Dourish, P. 2001. *Where the Action Is: The Foundation of Embodied Interaction*. Cambridge, Massachusetts: MIT Press.

Drummond, J. 2009. "Understanding Interactive Systems." *Organised Sound* 14(2):124–133.

Ebcioğlu, K. 1992. "An Expert System for Harmonizing Chorales in the Style of J. S. Bach." In M. Bablan, K. Ebcioğlu, and O. Laske, eds. *Understanding Music with AI: Perspectives on Music Cognition.* Cambridge, Massachusetts: MIT Press/AAAI Press, pp. 294–333.

Gaver, W. W. 1991. "Technology Affordances." *Proceedings of the 1992 ACM Conference on Human Factors in Computing Systems*. New York: Association for Computing Machinery, pp. 79–84.

Gibson, J. J. 1979. *The Ecological Approach to Visual Perception*. Boston, Massachusetts: Houghton Mifflin.

Grint, K., and S. Woolgar. 1997. *The Machine at Work*. Cambridge, UK: Polity Press.

Gurevich, M., and J. Treviño. 2007. "Expression and Its Discontents: Toward an Ecology of Musical Creation." *Proceedings of the 2007 International Conference on New Interfaces for Musical Expression*. New York: Association for Computing Machinery, pp. 106–111.

Hayles, N. K. 1999. *How We Became Posthuman.* Chicago, Illinois: University of Chicago Press.

Hunt, A., and R. Kirk. 2000. "Mapping Strategies for Musical Performance." In M. Wanderley and M. Battier, eds. *Trends in Gestural Control of Music*. Paris: IRCAM, pp. 231–258.

Hutchby, I. 2001. "Technologies, Texts and Affordances." *Sociology* 35:441.

Jordà, S. 2005. "Digital Lutherie: Crafting Musical Computers for New Musics' Performance and Improvisation." Ph.D. Thesis, University of Pompeu Fabra.

Kiefer, C., N. Collins, and G. Fitzpatrick. 2009. "Phalanger: Controlling Music Software With Hand Movement Using a Computer Vision and Machine Learning Approach." *Proceedings of the 2009 International Conference on New Interfaces for Musical Expression*. New York: Association for Computing Machinery, pp. 246–250.

Latour, B. 1994. "On Technical Mediation: Philosophy, Sociology, Genealogy." *Common Knowledge* 3(2):29–64.

Leman, M. 2008. *Embodied Music Cognition and Mediated Technology*. Cambridge, Massachusetts: MIT Press.

Magnusson, T. 2007. "The ixiQuarks: Merging Code and GUI in One Creative Space." *Proceedings of the 2007 International Computer Music Conference*. San Francisco, California: International Computer Music Association, pp. 332–339.

Magnusson, T. 2009. "Of Epistemic Tools: Musical Instruments as Cognitive Extensions." *Organised Sound* 14(2):168–176.

Magnusson, T., and E. Hurtado Mendieta. 2008. "The Phenomenology of Musical Instruments: A Survey." *eContact: Improv* 10(4). Available on-line at cec.concordia.ca/econtact/10_4/index.html. Accessed September 2009.

McLean, A. 2004. "Hacking Perl in Nightclubs." Available on-line at www.perl.com/pub/a/2004/08/31/livecode.html. Accessed June 2010.

Nilson, C. 2007. "Live Coding Practice." *Proceedings of the 2007 International Conference on New Interfaces for Musical Expression*. New York: Association for Computing Machinery, pp. 112–117.

Norman, D. A. 1988. *The Psychology of Everyday Things*. New York: Basic Books.

Norman, D. A. 1999. "Affordances, Conventions, and Design." *Interactions* 6(3):38–41.

Orio, N., N. Schnell, and M. M. Wanderley. 2001. "Input Devices for Musical Expression: Borrowing Tools from HCI." *Proceedings of the 2001 International Conference on New Interfaces for Musical Expression*. New York: Association for Computing Machinery. Available on-line at www.nime.org/2001/papers/orio.pdf.

Paine, G. 2009. "Towards Unified Design Guidelines for New Interfaces for Musical Expression." *Organised Sound* 14(2):142–155.

Pearce, M., and G. Wiggins. 2002. "Aspects of a Cognitive Theory of Creativity in Musical Composition." *Proceedings of the ECAI02 Workshop on Creative Systems*, Lyon. Available on-line at citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.7028.

Pérez, M. A., B. Knapp, and M. Alcorn. 2007. "Diamair: Composing for Choir and Integral Music Controller." *Proceedings of the 2007 International Conference on New Interfaces for Musical Expression*. New York: Association for Computing Machinery, pp. 27–28.

Polanyi, M. 1966. *The Tacit Dimension*. Garden City, New York: Doubleday.

Sorensen, A., and A. Brown. 2007. "aa-cell in Practice: An Approach to Musical Live Coding." *Proceedings of the 2007 International Computer Music Conference.* San Francisco, California: International Computer Music Association, pp. 292–299.

Varela, F., E. Thompson, and E. Rosch. 1991. *The Embodied Mind*. Cambridge, Massachusetts: MIT Press.

Vera, A. H., and H. A. Simon. 1993. "Situated Action: A Symbolic Interpretation." *Cognitive Science* 17:7–48.

Waisvisz, M. 1999. "Gestural Round Table." Available on-line at www.steim.org/steim/texts.php?id=4. Accessed September 2009.

Wanderley, M. M. 2000. "Gestural Control of Music." Available on-line at recherche.ircam.fr/equipes/analyse-synthese/wanderle/Gestes/Externe/kassel.pdf. Accessed September 2009.

Waters, S. 2007. "Performance Ecosystems: Ecological Approaches to Musical Interaction." *EMS: Electroacoustic Music Studies Network*. Available on-line at www.ems-network.org/spip.php?article278. Accessed June 2010.

Wessel, D., and M. Wright. 2001. "Problems and Prospects for Intimate Musical Control of Computers." *Computer Music Journal* 26(3):11–22.

Winkler, T. 1995. "Making Motion Musical: Gestural Mapping Strategies for Interactive Computer Music." *Proceedings of the 1995 International Computer Music Conference*. San Francisco, California: International Computer Music Association, pp. 261–264.

Winograd, T., and F. Flores. 1986. *Understanding Computers and Cognition*. Norwood, New Jersey: Ablex.