



Beyond the Keyboard and Sequencer: Strategies for Interaction between Parametrically-Dense Motion Sensing Devices and Robotic Musical Instruments

Jingyin He^{*†}, Jim Murphy^{*†}, Ajay Kapur[†], Dale A. Carnegie[†]

^{*}New Zealand School of Music, Victoria University of Wellington, New Zealand

[†]School of Engineering and Computer Science, Victoria University of Wellington, New Zealand

jon.he@vuw.ac.nz

Abstract

The proliferation and ubiquity of sensor, actuator and microcontroller technology in recent years have propelled contemporary robotic musical instruments (RMIs) and digital music controllers to become more parametrically dense than their predecessors. Prior projects have focused on creating interaction strategies for relatively low degrees-of-freedom input and output schemes. Drawing upon prior research, this paper explores schemes for interaction between parametrically-dense motion-based control devices and contemporary parametrically-dense robotic musical instruments. The details of two interaction schemes are presented: those consisting of one-to-one control (allowing the actions of a performer to directly affect an instrument) and those consisting of a recognition system wherein user-created gestures result in output patterns from the robotic musical instrument. The implementation of the interaction schemes is described, and a performance utilizing these schemes is presented.

Keywords: human-robot interaction, wearable sensor interfaces, musical robotics

1. Introduction

Contemporary robotic musical instruments (RMIs) have become more parametrically dense than their predecessors, with more actuators controlling greater numbers of musical parameters. As these new instruments present users with increasing numbers of actuators, they potentially afford composers and musicians greater expressive control than was possible with earlier systems. An increased number of parameters is accompanied by an increased level of difficulty in interfacing with them in real time: a system with many high-resolution output actuators will likely prove more time-consuming and difficult to interface with (in an intuitive and expedient manner) than a simpler RMI with few low resolution parameters. To address this problem, new map-

ping strategies between a human performer and a parametrically-dense RMI must be explored.

In attempting to create mapping strategies between human and robot, a challenge emerges: how to control what is essentially a complicated assembly of actuators in a manner as musically-sensible and relevant to a performer's skillset as possible. The goal of this paper is to explore this challenge, developing means of allowing the use of such complicated robotic instruments in performances through the use of parametrically-dense input interfaces. In doing so, a wearable input interface (designed by the first author) with many sensors is used to control a robotic output device with many actuators (designed by the second au-

thor). In the act of interfacing these two new I/O devices, a need was identified for a streamlined method of integration between them. To this end, new interfacing schemes were developed and a performance case study was undertaken. This work presents one of many possible mapping strategies that complicated I/O devices may employ.

Following a brief history of interaction schemes between human performers and RMIs, the hardware used in the performance case study is described. After introducing the hardware, the two human-robot performance paradigms explored in this paper are presented with implementation details, followed by a description of a performance featuring the hardware and software. Finally, this paper closes with a discussion of the work's outcomes and the potential for future work arising from that presented in this paper.

2. Background and Related Works

A challenge facing builders of robotic musical instruments is to create interaction schemes that allow users to create music with the instruments in a manner that affords creative exploration. The instruments themselves do not necessarily afford such interaction without significant intervention on behalf of the instrument builder: most of the actuators are electronically driven and numerically controlled. To enable musicians to use the instruments in a manner more similar to the instruments with which they are familiar, interaction schemes must be imposed upon the instrument. The work presented in this paper builds upon a history of such schemes.

Perhaps the fundamental way of defining a musical interaction paradigm for a musical robotic instrument lies in the mapping of the robot's outputs to a scheme based upon the MIDI keyboard control paradigm. In such systems, pitch-shifting elements are often assigned MIDI commands pertaining to pitch information. MIDI NoteOn and NoteOff commands may control actuation and damping mechanisms. In essence, the musical robot is mapped to respond to typical MIDI keyboard commands. While this system is widespread and

applied to many systems both simple and relatively complicated (exemplified in and (Focke, 2011) and (Singer et al., 2004)), it fails to account for the current trend in increasing parametric density on contemporary musical robots (Murphy et al., 2015). As musical robots gain more parameters, the number of relevant MIDI commands falls short. Further, attempting to map a robot's functionality to keyboard-oriented MIDI schemes neglects to account for those systems that contain expressive parameters quite different from keyboard-like instruments.

To address the shortcomings of the manual MIDI mappings presented above, a number of instrument builders have created mapping schemes for their systems that attempt to more closely match a particular robot's performance capabilities. A drum robot, for example, might have a percussion-based means of performer input. Notably, third author Ajay Kapur has created a mapping scheme between his *eSitar* and his *MahaDeviBot* (Kapur, 2008). This scheme, exemplified in the live performance *Digital Sankirna*, consists of a direct mapping between *eSitar*-mounted and wearable sensors and the robot's actuators (Kapur et al., 2011). During performance, the human user's motions explicitly trigger a musical robot output in a "one-to-one" mapping scheme: pressure on one of the *eSitar*'s frets, for example, results in the direct triggering of a robotic drum mechanism. While this scheme is used to control the relatively parametrically-simple *MahaDeviBot*, it allows a performer to escape the MIDI keyboard or sequencer paradigm typical of numerous other musical robots. Musical robot researcher Gil Weinberg, who has created a number of instruments that allow for unusual performer/robot interactions, exemplifies a second mapping paradigm. In (Weinberg and Driscoll, 2006), for example, a human-played drum pattern is input into a system to allow a musical robot to "improvise" in a manner based upon the human-input pattern. While sophisticated, this system (like Kapur's) consists of a parametrically-simple low-degree-of-freedom musical robot as the output device coupled to a low-number-of-inputs input device. As the complexity of the robotic instrument increases, the challenge of developing

musically-useful mapping schemes grows accordingly.

These three mapping paradigms encompass the majority of existent human/robot mapping schemes. While the MIDI keyboard-inspired scheme is used on numerous devices, it was deemed insufficiently applicable to contemporary parametrically-dense robot systems. The one-to-one performer/robot mapping and the higher-order mapping schemes mentioned in the preceding paragraph, however, are appropriate for new instruments with large numbers of controllable output parameters. These schemes are further explored below.

3. Interaction System Overview

With higher degrees of freedom afforded by new sensors, actuators and computing power, it is now possible to interact more intuitively with robotic musical instruments through musical gestures in both sonic and control paradigms. Two interaction schemes, based on musical gestures to control parametrically-rich RMI, were investigated in a performance context. The first interaction scheme utilizes the method of mapping sensor data of the input device into musical gestures. These gestures are then parameterized as instructions for the actuators of the RMI. The second interaction scheme involves the RMI playing a musical phrase (either stochastically generated or pre-composed) based on previously played phrases and user input. Following a description of the hardware and software used in this project, the subsequent subsections detail these interaction schemes.

3.1 *Kontrol*, *Tangle* and *Swivel 2*

The input interface used in this work is the *Kontrol* hardware and software system, while the output interface used is the *Swivel 2* mechatronic chordophone connected to the *Tangle* software suite (which allows for human-RMI interaction via MIDI). **Table 1** details the parameters of *Kontrol* and *Swivel 2*.

The *Kontrol* physical gesture acquisition system is a novel motion-based wearable

(seen in **Figure 1**) that measures finger flexion and physical dynamics of the hand (He, et al., 2015). It samples data at 100 Hz and sends it wirelessly to the laptop. The software processes and inputs the incoming sensor data to classifiers. The classifiers then categorize the hand postures and gestures of a Guqin (plucked 7-string Chinese instrument) performer in real-time.

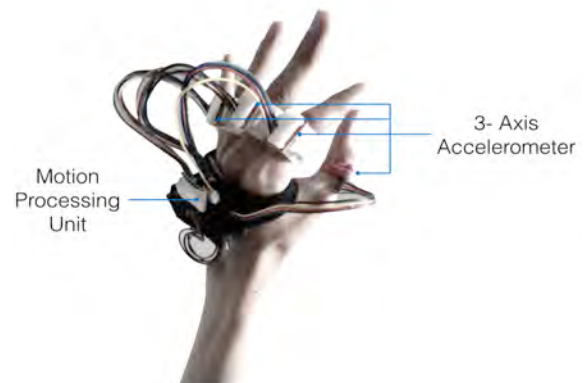


Figure 1. Kontrol wearable interface

Table 1. Summary of hardware parameters of Kontrol and Swivel 2

Kontrol Physical Gesture Acquisition System
Finger: 3-axis Accelerometer. Measures the flexion/extension of the fingers' metacarpophalangeal joint with respect to gravity.
Back of left palm: 9-axis Motion Processing Unit. Measures the pronation/supination, flexion/extension and ulnar/radial deviation of the left wrist, and acceleration with respect to gravity.
Xbee series 1 module. Transmits serial data to the host computer wirelessly.
Swivel 2 module (note: the complete instrument consists of 6 of these modules)
Fretter Arm Control: Miniature servo actuator. Positions fretter arm along string length for pitch shifting.
String picker: Miniature servo actuator. Rotates a guitar pick against the module's guitar string.
Clamper for fretter: Miniature servo actuator. Presses the fretter arm against the string, changing the string's pitch.
Damper for string: Miniature servo actuator. Rotates a foam pad against the vibrating string, dampening it.

Swivel 2 is a RM chordophone designed with high-resolution actuators that allow for the precise continuous control of the instrument's parameters. Illustrated in **Figure 2**, these parameters include as damping, fretting position, fretting depth, and picking (Murphy et al., 2013). *Swivel 2* is controllable through MIDI using the *Tangle* musical robot abstraction framework.

Tangle, described in more detail in (Mathews, et al., 2014), allows for user-defined MIDI, Open Source Control (OSC), or serial input to be mapped to RMI-specific output scheme. The *Tangle* framework is designed to be flexible regarding both input and output, and may be configured to listen for any input pattern and output in a scheme suitable for varying RMI communications methods. In the system described in this paper, *Tangle's* role is to provide an easy-to-use interface between the classification system and the high number of actuators present on *Swivel 2*, allowing for simplified communications between *Kontrol*-related software and *Swivel 2*-related software.

A typical data flow between *Kontrol* and *Swivel 2* is outlined in **Figure 3**. During a live performance, the performer wears *Kontrol*, which serially transmits the measured data to the host computer using wireless Xbee Series 1 modules. The gesture recognition system outputs hand posture classifications and motion data such as linear body acceleration and jerk without gravity. The output values are then sent via OSC to *Tangle*, in which the data is interpreted and converted to the MIDI-formatted data used to communicate with the microcontrollers responsible for the control of the actuators of *Swivel 2*. *Swivel 2* utilizes single-coil pickups (like an electric guitar) to capture the sounds produced and sends the audio signal back to the host computer through a Firewire audio interface for further processing. A chain of audio effects to effect the original audio is set up in either a DAW (such as Ableton Live) or a music programming language (Max/MSP or Chuck). Additionally, audio effects are controllable by the performer by mapping either the raw sensor data or the output of the gesture recognition system to the parameters of the audio effects.

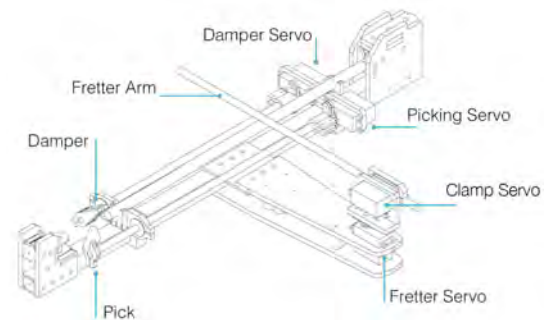


Figure 2. Module of *Swivel 2* (the complete instrument consist of 6 modules)

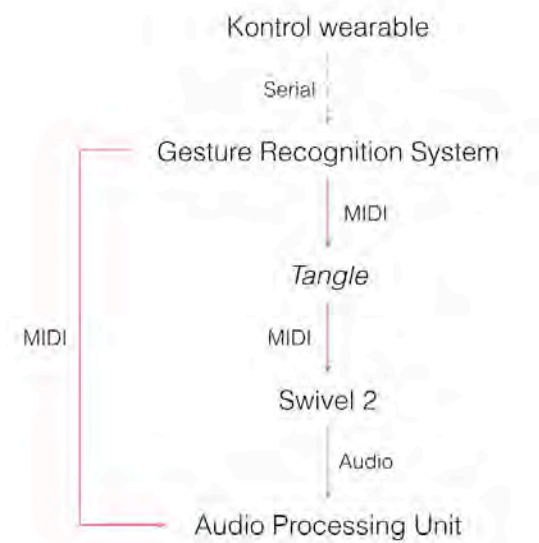


Figure 3. Data flow of interactive system

3.2 Connecting the dots between *Tangle* and *Kontrol* Gesture Recognition System

As mentioned in earlier sections, the interaction schemes deployed aim to explore the interaction paradigms beyond the traditional one-to-one mapping schemes between performer and robot.

In the first interaction scheme (IS-1), abstractions of sensor data as hand postures and gestures from the *Kontrol* system are mapped to the abstractions of actuating mechanisms of *Swivel 2* through *Tangle*. This mapping scheme addresses the parameterization between abstractions at the *sound object* time scale, whereby sonic events with duration from a fraction of a second to several seconds occur (Roads, 2001). A general mapping scheme between abstractions is as follows: If

Posture A is active, then n numbers of strings (up to six) are fretted at $f_1, f_2... f_n$ position(s) on string(s). An example would be that a clenched first gesture could lead to all strings being fretted with depth determined by wrist flexion, and with each of those being fretted at different points based on the shape of instantaneous finger flexion data. While the wrist radial/ ulnar deviation offsets each of the fretter positions, high jerk in wrist motion would result in the excitation of the strings by the RMI's string picker. **Table 2** shows a summary of the mapping used in IS-1.

Table 2. General scheme of parameterizing N number of postures to actuating Swivel 2

Kontrol	Tangle + Swivel 2
Hand posture [0]	No strings fretted
Hand posture [1] Shape of instantaneous finger flexion data Wrist flexion	1 string fretted, 4 strings touching Fretter position Fretting depth
Hand posture [2] Shape of instantaneous finger flexion data Wrist flexion	2 string fretted, 3 strings touching Fretter position Fretting depth
...	...
Hand posture [N] Shape of instantaneous finger flexion data Wrist flexion	N strings fretted, (5 - N) strings touching Fretter position Fretting depth
Wrist ulnar/ radial deviation	Offset of fretter position

In interaction scheme 2 (IS-2), the user's hand postures and gestures are mapped to control *Swivel 2* in the *meso* timescale, whereby phrases of musical/ sonic structures of various lengths measured in minutes or seconds occur (also presented in (Roads, 2001)). Each posture and gesture is mapped to trigger a sequence, while their motion data are used to alter the parameters of the sequence such as playback speed. Additionally, the postures and

gestures also progress a score (similar to that of Max Mathew's score following with the Radio-Baton (Mathews, 1989)). The score following system used here is capable of branching, allowing for nonlinear compositional progression at the performer/composer's discretion as seen in **Figure 4**. The pre-composed variable-length sequences are grouped according to their sonic characteristics, and the sections' labels do not follow an order of progression. Each variation in each of the sections can be accessible from any section based on the requirements set for each sequence. The requirements are based on how many performed postures and/or gestures have been detected. Along with the requirement threshold for triggering each sequence, the histogram of performed postures and gestures is reset each time a new sequence is triggered (regardless of whether it is within or beyond the section).

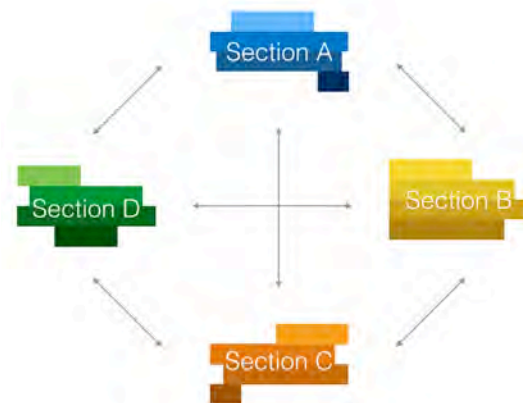


Figure 4. A network of sequences for non-linear compositional progression.

4. Applications

The interaction system presented above premiered at the solo recital performance of acclaimed Guqin performer Wu Na at the Adam Concert Room, New Zealand School of Music. The main objectives of the performance were to: 1) demonstrate the system outside of the laboratory in a live performance situation, 2) investigate the immediacy of the integration of the interaction system, and 3) explore the sonic possibilities of the presented human-robot interaction schemes in improvisatory music performance. The instrumentalist did

not have any prior rehearsals with the system, although there had been discussion about the general form of the piece.



Figure 5. Wu Na, during IS-2, waiting for *Swivel 2*'s response (<https://vimeo.com/129071867>).

The piece had two distinct movements characterized by its musical outcome influenced by the interaction system employed. The first movement was presented as introduction and exposition. During this movement, *Swivel 2* was controlled using the traditional interaction methods of on-the-fly triggering and editing pre-composed sequences. The implication of this model of interaction in musical outcome was that the human performer was mostly leading the improvisation, while *Swivel 2* assumed a more supportive role. In the second movement, the Guqin performer put on the *Kontrol* interface as *Swivel 2* finished the remainder of its current ongoing sequence. The new interaction schemes were then deployed, and the roles of the performers (human and robot) were observed to evolve during the second movement. With IS-1 in place, the performer was observed to intuitively dictate *Swivel 2*'s output through the use of both traditional and non-repertoire playing techniques. Conversely, with IS-2, *Swivel 2* is observed to be more autonomous, leading to a more dynamic conversation with the Guqin performer - influencing, complementing and contrasting the musical intentions of the human performer. The interaction in this case is bi-directional, resulting in tension and release, as well as complimentary and contrasting human-robot interaction. This is observed to a degree not present in IS-1.

Through this performance, it is observed that the three aforementioned objectives were satisfactorily met. In meeting the first objective, the system performed in a stable, glitch-free manner during a real-world performance scenario. In providing this system to a performer with little prior knowledge of the apparatus, a compelling and sonically-interesting live improvisation was performed in a manner that complimented the repertoire of both traditional and extended techniques of Guqin performance. This indicates that both the second and third goals of the *Kontrol-Swivel 2* interaction schemes were also met.

5. Conclusion and Future Works

When attempting to control a densely-parameterized musical robot with a parametrically-rich output interface, a difficulty emerges in mapping the output device's values to the input parameters of the robot. The work presented in this paper has sought to address this challenge through the use of a number of interaction strategies, presenting new abstraction and mapping schemes for live performance use.

With the interaction schemes presented in this paper, a performer may gain a level of real-time access and control over the RMI that they may not have had they been required to manually input each actuator control value. Indeed, where many prior musical robots are essentially treated as output devices to be connected to traditional sequencers, the interaction strategies presented herein allow musicians to gain direct and musically-relevant control over and otherwise-complicated instrument. In addition to providing musicians with this musically-relevant control, gesture-to-sound schemes provide audiences with an overview of performative cause and effect between musician and digital instrument that is difficult to achieve within the aforementioned sequencer-like paradigm.

Though the work presented in this paper has focused on a particular performance paradigm, it is hoped that it is merely the first in a series of works featuring novel output devices controlled by novel input devices. Such would

serve to bridge the authors' research group's creative avenues. To go toward allowing any new musical output interface to control any new RMI, the Tangle musical robotic framework must be further extended to be applicable to any new devices. After extending Tangle, the creation of new musically-relevant mapping strategies may be explored. Additionally, *Kontrol* hand gesture recognition framework should also expand its recognition framework to other instruments such that performance techniques of other instrumentalists or even dancers may be explored.

References

- Focke, A. (2011). *Trimpin: Contraptions for Art and Sound*. Marquand Books, Seattle, Washington, 1st edition.
- He, J., Kapur, A., and Carnegie, D. (2015). Developing a physical gesture acquisition system for Guqin performance. *Proceedings of 2015 International Conference on New Interfaces for Musical Expression*. Baton Rouge, Louisiana.
- Kapur, A. (2008). *Digitizing North Indian Music: Preservation and Extension using Multi-modal Sensor Systems, Machine Learning and Robotics*. VDM Verlag.
- Kapur, A., Darling, M., Diakopoulos, D., Murphy, J., Hochenbaum, J., Vallis, O., and Bahn, C. (2011). *The machine orchestra: An ensemble of human laptop performers and robotic musical instruments*. *Computer Music Journal*, 35(4):1–15.
- Mathews, M. V. (1989). The conductor program and the mechanical baton. In M. Mathews & J. Pierce (Eds.). *Current Directions in Computer Music Research* (pp. 263-282). Cambridge, MA: The MIT Press.
- Mathews, P., Murphy, J., Carnegie, D., Kapur, A. (2014). Tangle: a flexible framework for performance with advanced robotic musical instruments. In *Proceedings of the 2014 Conference on New Interfaces for Musical Expression*. London.
- Murphy, J., Kapur, A., and Carnegie, D. A. (2013). Swivel 2: A systems overview and evaluation of a new mechatronic slide guitar. In *Proceedings of the 2013 Engineering New Zealand Conference (ENZCON)*, Auckland, New Zealand.
- Murphy, J., McVay, J., Mathews, P., Carnegie, D., Kapur, A. (2015). Expressive Robotic Guitars: Developments in Musical Robotics for Chordophones. *Computer Music Journal* 39(1), 59-73.
- Roads, C. (2001). *Microsounds*. The MIT Press.
- Singer, E., Fedderson, J., Redmon, C., and Bowen, B. (2004). Lemur's musical robots. In *Proceedings of the 2004 Conference on New Interfaces for Musical Expression (NIME)*, Hamamatsu, Japan.
- Weinberg, G. and Driscoll, S. (2006). Toward robotic musicianship. *Computer Music Journal*, 30(4):28–45.