



UNIVERSITY
OF WOLLONGONG
AUSTRALIA

University of Wollongong
Research Online

Faculty of Creative Arts - Papers (Archive)

Faculty of Law, Humanities and the Arts

2005

Pocket Gamelan: A Blueprint for Performance Using Wireless Devices

Greg Schiemer

University of Wollongong, schiemer@uow.edu.au

Mark Havryliv

University of Wollongong, mh675@uowmail.edu.au

Publication Details

This conference paper was originally published as Schiemer, G and Havryliv, M, Pocket Gamelan: A Blueprint for Performance Using Wireless Devices, in free sound ICMC 2005 - Proceedings of the International Computer Music Conference, Barcelona, Spain, 5 - 9 September 2005, 600-603.

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library:
research-pubs@uow.edu.au

POCKET GAMELAN: A BLUEPRINT FOR PERFORMANCE USING WIRELESS DEVICES

Greg Schiemer

Faculty of Creative Arts
University of Wollongong
NSW Australia 2522
{schiemer@uow.edu.au}

Mark Havryliv

Faculty of Creative Arts
University of Wollongong
NSW Australia 2522
{mh675@uow.edu.au}

ABSTRACT

Mobile phone handsets have introduced new possibilities for musical interaction between multiple performers, as we reported in previous papers. Wireless communication between handsets now extends these possibilities even further. This paper describes development and implementation of a new performance scenario that involves remote instrument control using a Bluetooth connection. The paper proposes a low-level functional control protocol designed primarily around the current state of the mobile phone handset. The protocol makes provision for extended musical functionalities developed around tuning systems that are not adequately served by existing musical performance interfaces based on twelve equal divisions of the octave. Development is also driven by enhancements in handset technology as new hands-free devices are developed. The way is also left open to include Bluetooth wireless communications protocols in a wide variety of musical applications that take advantage of the mobility offered by personal networks and enable new kinds of musical interactivity.

1. INTRODUCTION

Wireless networks have opened up possibilities for designing a new generation of interactive musical instrument. Such an instrument is hand-held and can operate without signal cables. This has potential to extend ways in which musicians interact with one another. Appropriate metaphors for realising the musical potential of this technology may be found by observing the dynamics of ensemble playing.

In orchestral, choral or chamber music, individual players affect the ensemble sound in several ways. The player not only responds to the sound emanating from his/her instrument but also receives and responds to auditory feedback from other members of the ensemble. The integrity of ensemble playing usually depends on the collective rhythmic and intonational response of each player to the ensemble sound produced. This property is the result of a complex system of auditory and neurological feedback that operates between every player. The flow of information happens in realtime and is distributed among all members of the ensemble.

Such a collaborative metaphor is a radical departure from the underlying metaphor that computer music inherited from Music4 which focused on a different model of music production, namely composers creating scores for orchestras. Within the desktop computing

environment where the legacy of Music 4 can still be found, control of the sound of each instrument is concentrated in the hands of a single user and processed on a single machine.

The mobility offered by wireless technology has introduced unprecedented possibilities for musical interaction between members of an ensemble. As this technology becomes integrated with new consumer products it provides a framework that allows computer musicians to rethink musical strategies. By observing the musical interaction that takes place in traditional ensemble playing it is possible to gain an understanding of ad hoc networks and realise their full musical potential.

2. POCKET GAMELAN

The Pocket Gamelan is a set of mobile musical instruments. ‘Gamelan’, in the title, is a metaphor for the kind of musical interaction (discussed above). The project seeks to develop a software prototype for a network of hand-held instruments. Instruments are configured in a network in which applications software is distributed between a web server and multiple clients.

The instruments of the Pocket Gamelan have been appropriated from mobile phone technology. Each mobile unit is easy to play, quick to learn and produces audible tones that are microtonally tunable. Each unit is battery powered and able to take advantage of new developments in mobile digital computing. Each unit is a hand-held sound source that is played by pressing buttons and can also be extended to include a variety of movement sensors that provide a wearable performance interface. Players are free to move each sound source while performing.

The principal motivation for the project has been a desire to develop an extensible interface that supports performance of music in various tuning systems free of the tuning constraints associated with conventional music performance interfaces. The diversity of available tuning systems, as reflected in the theoretical writings of Partch [1], Chalmers [2] and Wilson [3, 4, 5, 6, 7], calls for such an extensible interface.

In previous papers, we described two performance scenarios for the Pocket Gamelan in which mobile handsets can be used in performance [8, 9, 10]. While these performance scenarios were also based on group musical interaction, each mobile phone unit was played as a stand-alone device; each unit was autonomous rather than networked and did not depend on control

being communicated during performance between one handset and another.

2.1. Java development

Applications for these scenarios were developed in J2ME using Eclipse and Java Wireless Toolkit and tested using a Nokia 6230 mobile phone handset. Applications were downloaded into the handset using a USB cable.

The new scenario discussed here takes advantage of the fact that the Nokia 6230 is Bluetooth-enabled. This allows the volume and pitch of a moving sound source to be modified by remote control. It also allows multiple parts played on different handsets to be synchronised or played via manual operations performed on a remote handset.

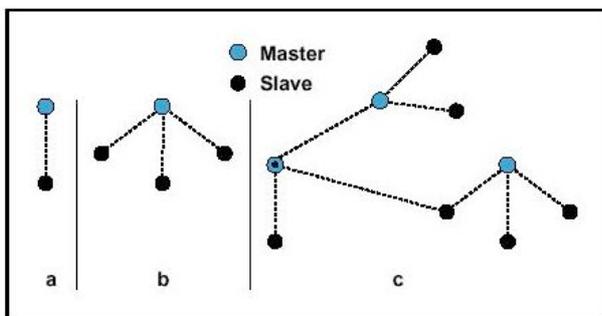
3. AD HOC NETWORKS

For every Bluetooth link between two or more devices, one device is defined as master and the others as slaves. This forms a piconet which can contain up to seven devices. A device can act as both a master in one piconet and a slave in another, joining two piconets, creating a scatternet [12].

Two-way communication between pairs of musicians in a musical ensemble may be compared to the point-to-point connection between a master and a slave device in a Bluetooth piconet, as shown in Figure 1 (a).

Alternatively, just as a single instrumentalist can provide rhythmic or motivic cues for several players in the ensemble, the piconet allows a master to transmit cues simultaneously to several other slave devices, as shown in Figure 1 (b).

Alternatively, just as a single instrumentalist can respond to several rhythmic or intonational cues received simultaneously from players in different sections of the ensemble, the piconet allows a slave device to respond to simultaneous cues sent from different masters. Master devices scattered throughout different parts of the network may transmit cues simultaneously to several other slave devices, as shown in Figure 1 (c).



a) Point-to-point connection between two devices
 b) Point-to-multipoint connection between a master and three slaves
 c) Scatternet that consists of three piconets [Core, p.42]

Figure 1. Different kinds of networks and Master/Slave relationships are shown in Forum Nokia's Bluetooth Technology Overview, p.7.

Like hot-pluggable networks (e.g. Firewire, USB), arbitration associated with reconfiguration takes place automatically. Users of musical applications are expected to be unaware of arbitration taking place in the Bluetooth environment.

3.1. Bluetooth protocol stack

The Bluetooth protocol stack describes the hardware and software layers that give an application access to wireless communication. A simplified summary of the protocol is found in Forum Nokia's Bluetooth Technology Overview [13].

The function of each protocol is now described and illustrated, where possible, using musical analogies.

3.1.1. Device Enquiry

The Device Enquiry Protocol defines the process whereby devices in a wireless network are able to locate, identify and communicate with other devices nearby. The protocol supports three states in which each device may be found by other devices. Each device is initialised in one of the following states:

- General Unlimited Access Inquiry Code (GIAC) – allows a device to be found for an unlimited time;
- Limited Dedicated Access Inquiry Access Code (LIAC) – allows a device to be found for a limited time only, typically one minute; and
- Not discoverable [14, 15, 16].

Unlimited access (GIAC) may be appropriate in a gallery setting when mobile devices are used by the public as a user interface for an interactive sound installation.

Limited access (LIAC) may be appropriate in a performance setting when devices are used interactively by several performers. Prior to the performance each device must establish network connections. While the performance is underway, these devices may then revert to the 'not discoverable' state to exclude unwelcome interaction from uninvited participants.

3.1.2. Service Discovery

Before a device can connect to another device on the network, it must first determine whether there is a remote device present that will provide the service it requires.

For example, as part of the Service Discovery Protocol (SDP), a device, e.g. a printer, may first notify any potential master device that might attempt to make a connection that it is a printer before allowing connection to be made. This is quite different to MIDI which has limited means of knowing whether or not a particular target device is present on the daisy-chain.

A device may also offer more than one service and be aware of which service is being requested at any time.

3.1.3. Universally Unique Identifiers

When a service discovery is performed, a list of Universally Unique Identifiers is returned and compared to a local list to find and connect to a desired service.

A Universally Unique Identifier (UUID) is a 128-bit value that is guaranteed to be unique across all space and time. It identifies the services a device supports.

3.1.4. Name Discovery

When one device identifies another, a remote device will make simple requests of the device found. These requests may include the device's 'friendly' name, its Bluetooth address, its authentication status or a change in its encryption setting. A 'friendly' name can be designated instead of a UUID. Using a 'friendly' name simplifies service requests because users are not required to use a 128-bit UUID.

3.1.5. L2CAP

Logical Link Control and Adaptation Protocol (L2CAP) is the lowest-level Bluetooth communication layer accessible to a J2ME application. Other protocols that handle data in various formats operate via L2CAP. These include RFCOMM, SDP and TCS-binary.

While existing java APIs [17, 18] allow synchronous transmission of large data packets there are noticeable latencies when these are used to transmit small message packets such as MIDI messages.

On top of the L2CAP layer, we have found it necessary to create a customised API for Asynchronous Input Output – shown as AIO in Figure 2. This was necessary to accommodate smaller message packets within the Bluetooth Protocol. This API supports asynchronous byte-oriented protocols and supports MIDI message packets where data is sent in packets of 1, 2 or 3 bytes.

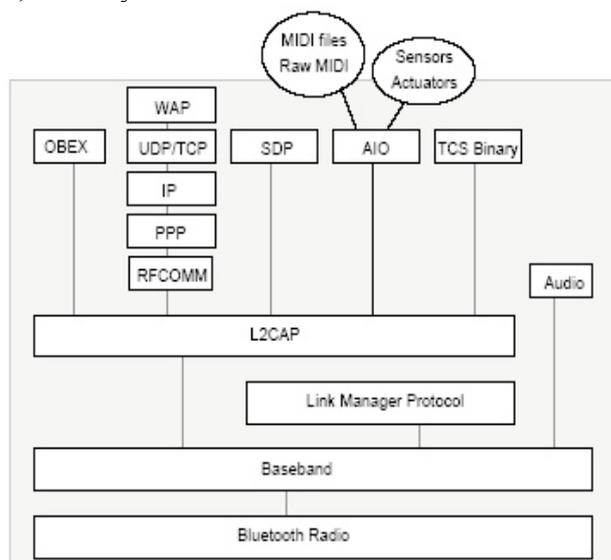


Figure 2. The Bluetooth Protocol Stack and external input and output devices. AIO is an API developed for Asynchronous Input and Output.

In a performance using mobile phones as the only sound generators, a single-byte operation by one performer may actively modify sounds created by another. This allows a performer to (e.g.) modify the pitch or amplitude envelope or a tuning algorithm running on a remote device.

3.1.6. Authentication Encryption and Authorization

Bluetooth security requires Authentication and Authorization of devices and services and encryption of data transferred. Authentication and Authorization takes place during Device Enquiry and Service Discovery. Encryption happens whenever data is received or transmitted. In general these security requirements are serviced in a layer that the user does not need to access.

In a musical performance scenario involving wireless communication, security is critical so that a performance or sound installation is not vulnerable to attack by 'bluesnarfers' or 'bluejackers'.

Mobile phone communications has another interesting musical legacy that predates the age of digital electronics. The frequency hopping algorithm which is used in mobile phones to minimise interference between communications channels first originated in a musical composition. A system developed by composer George Antheil to synchronise multiple player pianos in his 1925 composition *Ballet Mecanique* was later adapted as a secret communications device. A patent was granted to the composer and Hedy Lamarr in 1942 [11].

4. PERFORMANCE SCENARIOS

As more mobile phones, laptops and PDAs increasingly become Bluetooth enabled, musicians will use this technology in conjunction with legacy electronic music equipment. Miniature accelerometers that are Bluetooth enabled can add a further dimension of expressive performance control.

Using wireless dongles that plug into MIDI and audio sockets musicians will be able to perform on existing electronic musical instruments without the constriction of signal and power cables. Applications will be driven from a pre-programmed menu or controlled by a performer in real-time.

Some of the possibilities include:

- mobile handsets that control sound-generating devices via a wireless connection;
- wireless controllers that send control commands to desktop computers or audio/MIDI effects units; or
- wireless devices that route audio or MIDI signals between set-top devices.

Wireless accelerometers can be attached to the performer's body without cables that would otherwise constrict a performer's movements. To do this a number of dongles are being built. Each dongle will consist of a Bluetooth transceiver, a microcontroller, and will have either a male 5 pin DIN or a male mono audio connector. It will typically be a slave device in any Bluetooth performance piconet. Alternatively, it will have the capacity to act as a repeater node in a scatternet thereby extending the range of the wireless network as shown in Figure 3.

The number of devices that can be added to a network is not restricted by the number of physical sockets as is the case with wired devices. Devices can be placed up to 10 metres apart. This allows cabled

equipment to be relocated off stage keeping the stage area clear for the dramatic action of the performance.

Figure 3, shows a performer playing a MIDI keyboard wearing Bluetooth enabled accelerometers. The performer has an accelerometer dongle on each elbow and one on the head. Each sensor communicates wirelessly with a mobile phone. In addition, the keyboard transmits MIDI to the mobile phone via a MIDI dongle. The phone receives and translates sensor data before relaying MIDI commands to a MIDI-controlled synthesis application running on a laptop and a MIDI-controlled audio effects unit.

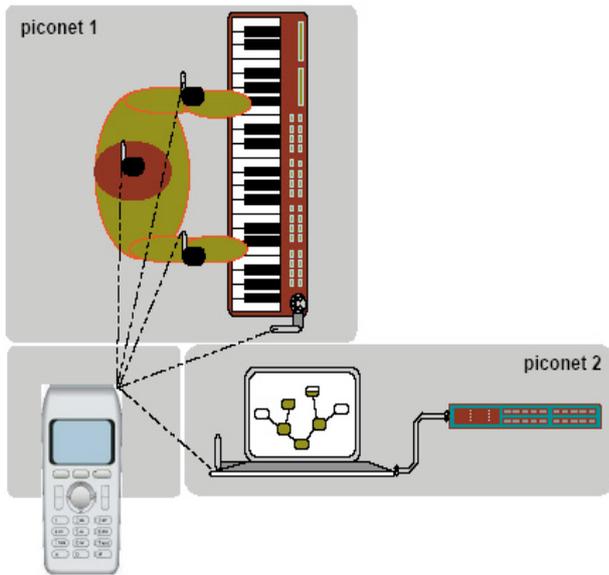


Figure 3. A performance scatternet formed by two piconets with the mobile handset is common to both piconets.

5. CONCLUSION

Instruments based on Bluetooth technology offer a degree of mobility and autonomy that conventional instruments give to musicians. The extent to which this technology affects performance of music is limited only by the ways in which performers are allowed to move as part of the performance and the kinds of spaces where a performance is presented.

Several performances have already been scheduled using the Pocket Gamelan in public concerts. These include *Mandala 3* performed Wednesday July 13th, Kelvin Grove, Brisbane, Australia and *Mandala 4* at UK Microfest, Saturday October 15th, UK Microfest, Riverhouse, Walton-on-Thames, England. Both works use Bluetooth-enabled phones in the simplest of wireless scenarios described above.

6. ACKNOWLEDGEMENTS

The Pocket Gamelan project is funded by a three-year grant from the Australian Research Council.

7. REFERENCES

- [1] Partch, H. *Genesis of a Music* (2nd ed.) Da Capo Press, New York, 1979
- [2] Chalmers, J. *Divisions of the Tetrachord* Frog Peak, New Hampshire, 1993
- [3] Wilson, E. Musical Instrument US Patent Office Patent Number 3,012,460 1961
- [4] Wilson, E. Musical Instrument Keyboard US Patent Office Patent Number 3,342,094 1967
- [5] Wilson, E. "The development of Intonational Systems by Extended Linear Mapping", *Xenharmonikôn 2* 1975a
- [6] Wilson, E. "Bosanquet – A Bridge – A Doorway to Dialog" *Xenharmonikôn 3* 1975b
- [7] Wilson, E. "D'Alessandro Like a Hurricane" *Xenharmonikôn 9* 1986
- [8] Schiemer, G., Sabir, K. and Havryliv, M. "The Pocket Gamelan: A J2ME Environment for Just Intonation" *Proceedings of the International Computer Music Conference*, Miami, USA, 2004
- [9] Schiemer, G. and Havryliv, M. "Wearable Firmware: The Singing Jacket" *Proceedings of Australasian Computer Music Conference*, Wellington, New Zealand, 2004
- [10] Schiemer, G., Alves, W., Taylor, S. & Havryliv, M. "Pocket Gamelan: Building The Instrumentarium For An Extended Harmonic Universe" *Proceedings of the International Computer Music Conference*, Singapore, 2003
- [11] Price, R: Further Notes and Anecdotes on Spread-Spectrum Origins see IV. Shortly before Pearl Harbour: The Lamarr-Antheil Frequency-Hopping Invention, *IEEE Transactions on Communications* Vol. COM-31, No.1 pp. 89-91 1983
- [12] Java Community Process *JSR-000082 Java APIs for Bluetooth*
<http://jcp.org/aboutJava/communityprocess/final/jsr082/index.html> 2002
- [13] Nokia *Bluetooth Technology Overview*
<http://forum.nokia.com/main.html> 2004
- [14] Mahmoud, Q. H. *Learning wireless Java*. O'Reilly, Sebastopol, CA, 2002.
- [15] Mahmoud, Q. H. *Wireless Application Programming with J2ME and Bluetooth*
<http://developers.sun.com/techttopics/mobility/midp/articles/bluetooth1/> 2003
- [16] Mahmoud, Q. H. *The Java API's for Bluetooth Wireless Technology*
<http://developers.sun.com/techttopics/mobility/midp/articles/bluetooth1/> 2003
- [17] Nokia *Introduction To Developing Networked MIDlets Using Bluetooth*
<http://forum.nokia.com/main.html> 2004
- [18] Sony Ericsson *Developing Applications with the Java APIs for Bluetooth (JSR-82)*
<http://www.microjava.com/articles/Bluetooth-jsr-82-training.pdf> 2004
- [19] Schildt, H. *Java 2: A Beginner's Guide*. McGraw-Hill/Osborne, New York, 2003.