Signal Analysis in Interactive Real-Time Music Systems

Si Waite

Noisefloor 28.04.2015
Contents

• Overview and context
• Why signal analysis?
• Scope
• Problems and solutions
• Review of techniques
• Next steps
• References
Interactive Real-Time Systems: Definition

“An interactive composing system operates as an intelligent instrument – intelligent in the sense that it responds to a performer in a complex, not entirely predictable way, adding information to what a performer specifies and providing cues to the performer for further actions...The computer responds to the performer and the performer responds to the computer, and the music takes its form through that mutually influential, interactive relationship.”

(Chadabe, 1984)
Interactive Real-Time Systems: Definition

“...systems on which we hear the result of our decisions as we take them. A traditional musical instrument is a ‘real-time system’.”

(Wishart, 1994)
Interactive Real-Time Systems: Rationale

• Allow the work to be recreated in the moment (Wishart, 1994)
• Increase accessibility to music-making (Machover, 1999)
• Facilitate listening in composition (Machover, 1999)
• Allow for multi-timbral control (Eigenfeldt, 2009)
• Increased range of expression (Reas & Fry, 2007)
• Composition as a process not a product (Nyman, 1999)
Interactive Real-Time Systems: Rationale

“...good things often happen – in work, in romance, and in other aspects of life - as a result of successful interaction during opportunities presented as if by chance...it seems to me reasonable that such a perception should also find expression in music”.

(Chadabe, 1984)
Context (My Research)

- Co-performers
- Exploration of chaos within parameters of popular song forms
- Varying levels of interactivity (varies at composition and performance stages)
- The human performer and system to be responsible for both high and low-level parameters (unlike conductor models)
Why Signal Analysis?

• New possibilities in timbral processing

  – Early systems were MIDI / control-data based (Eigenfeldt, 2007)

  – More powerful computers have opened up the possibility of more studio-based processes on the stage (Eigenfeldt, 2007)
Why Signal Analysis?

• Advantages of the instrument as controller
  – Minimises additional equipment (Richards, 2006)
  – Minimises additional performance demands
  – Allows co-performance of human and system
  – Non-invasive (Kristensen, 2012)
Why Signal Analysis?

• Range of features that can be detected

“The last 20 years have seen...the development of algorithms capable of onset detection, beat tracking, pitch detection, downbeat detection, chord recognition and many other forms of musical audio analysis.” (Stark, 2014)

Why Signal Analysis?

• Additional possibilities with further processing
  
  – Score-following (Winkler, 2001; Waite, 2014)
  
  – Pattern recognition using Machine Learning techniques (Caramiaux & Tanaka, 2013)
Scope

- The artist as programmer
- Max externals, VSTs or easily-configurable standalone software
- Use of transparent mappings
- Non-invasive techniques
- Portability
Problems & Solutions

• Accuracy vs Latency
  – Hardware alternatives (Pardue et al. 2014)
  – New algorithms (Kristensen, 2012)
  – Combining approaches
    • Multimodal approaches (Wishart, 1994)
    • Input filtering and additional processing (Stark, 2014)
  – Adapting the system
  – Audio analysis is never perfect (Jam Origin, 2014)
Problems & Solutions

• Additional control

“Open air gestures are not traditionally associated with music-making, offering the performer an opportunity to transcend habitual movement patterns and explore fresh links between gesture and sound.”

(Mainsbridge & Beilharz, 2014)
Problems & Solutions

• Computational load
  – Use of multiple computers and wireless communication protocols (Stark, 2014)
  – Development of new, more efficient techniques (Kristensen, 2012)
  – Don’t use Max...
Review of Techniques

Following tables and Max Patch detail:

- Analyzer~
- Gbr.yin
- Pipo~
- Zsa.Descriptors
- Retune~
- Sound Analyser
- Jam Origin
<table>
<thead>
<tr>
<th>Name</th>
<th>Hosting</th>
<th>Features</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyzer~ (Johan, 2001)</td>
<td>Max</td>
<td>Pitch detection: fundamental and partials</td>
<td>Lots of features combined into 1 external Low on higher pitches</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMS values of frequency components</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Loudness, Brightness, Noisiness, Bark Attack</td>
<td></td>
</tr>
<tr>
<td>Gbr.yin~ (Schnell &amp; Schwarz, 2005)</td>
<td>Max (Ftm&amp;Co)</td>
<td>Pitch detection</td>
<td>Very reliable, especially when used on bass notes with</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMS</td>
<td>low-pass filtering</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quality factor (periodicity)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Autocorrelation</td>
<td></td>
</tr>
<tr>
<td>Pipo slice:yin (Françoise et al, 2014)</td>
<td>Max (MuBu)</td>
<td>Pitch detection</td>
<td>Not yet tested</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RMS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Quality factor (periodicity)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Autocorrelation</td>
<td></td>
</tr>
<tr>
<td>Zsa.descriptors (Malt &amp; Jourdan, 2008)</td>
<td>Max (in pfft~)</td>
<td>Pitch detection: fundamental, virtual fundamental</td>
<td>Basic tests suggest unstable pitch tracking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Amplitude/emergy detection</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Spectral: Bark, Mel, Centroid, Flux &amp; more</td>
<td></td>
</tr>
</tbody>
</table>
## Signal Analysis Methods Overview (2)

<table>
<thead>
<tr>
<th>Name</th>
<th>Hosting</th>
<th>Features</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retune~ (Cycling 74, 2014)</td>
<td>Max 7 / M4L</td>
<td>Pitch detection, RMS</td>
<td>Additional creative possibilities</td>
</tr>
<tr>
<td>Sound Analyser (Stark, 2014)</td>
<td>VST and OSC</td>
<td>Time domain: RMS, Peak, Zero Crossing Rate</td>
<td>Can’t configure algorithms</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Frequency domain: Spectral Centroid, Flatness and Crest.</td>
<td>Unstable on pitch detection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Onset detection: Energy Difference, Spectral Flux, High Frequency Content</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>and Complex Spectral Difference.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Pitch detection: Yin, Chord recognition Spectra: FFT Magnitude, Mel-frequency representations and the Constant-Q</td>
<td></td>
</tr>
<tr>
<td>MIDI Guitar (NB not free!)</td>
<td>VST / Standalone</td>
<td>MIDI: Pitch, Velocity, Pitch Bend Monophonic / Polyphonic</td>
<td>Very reliable</td>
</tr>
<tr>
<td>(Jam Origin, 2014)</td>
<td></td>
<td></td>
<td>Saves a lot of work</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Highly configurable.</td>
</tr>
</tbody>
</table>
Next Steps

• Further testing
• Linking to processing
• Transparent mappings to generative processes
• Transparent mappings to audio and visual outputs
• Composing and performing


References


References


