P[a]ra[pra]xis: Poetry in Motion

Josh Dubrau
Faculty of Arts and Social Sciences
University of New South Wales
Australia
joshninelives@internode.on.net

Mark Havryliv
Sonic Arts Research Network
Faculty of Creative Arts
University of Wollongong
Australia
mhavryliv@gmail.com

ABSTRACT

P[a]ra[pra]xis is an open two-part software suite and Java library (JAR) that facilitates the realtime creation and simultaneous sonification of poetry/prose. It is particularly designed to implement word substitutions based on the psychoanalytical principles of free association and metonymic slippage.

The first part, *P[a]ra[pra]xis Collection Editor*, allows a user to create and maintain a dictionary of words and their grammatical properties (i.e. verb, singular noun, pronoun etc.) and the corresponding properties of user-defined substitutions for those words. The second part, *Realtime P[a]ra[pra]xis*, executes these substitutions as the user/performer types, and broadcasts OSC messages containing the properties of the original and substituted words, along with discrete notifications of keyboard events.

A case study (based on a live networked performance) is presented which highlights one particular usage of this program in the form of an Instant Messenger (IM) style chat with interpolated 'Freudian slips' to create a dialogue which changes between the point of transmission and the point of reception, and spontaneously generates music reflecting physical and emotional changes in the dialogue.

Keywords

Poetry, language sonification, psychoanalysis, linguistics, Freud, realtime poetry.

1. INTRODUCTION

Text-to-sound converters are not uncommon. Realtime music software like pd, Csound and SuperCollider can receive discrete keyboard events when a key is typed. Other software maps text (as ASCII characters) either to MIDI note numbers or to an MP3 file, invariably based on transmogrifications of alphabet positioning to pitch, texture or rhythm. More advanced converters create meta-descriptors (which may be based on a readability index, or some other lingual parser) which are then used to control musical parameters. Please see [1] for an extensive listing and discussion of software.

At the other end of the spectrum, sonification mechanisms have been developed that can be linked to specialist language systems. SoniPy[2] is an open framework for sonification written in Python, and can therefore be linked to the Natural Language Toolkit (NLTK) [3]. In turn, NLTK can import data and functions from Wordnet – a dictionary and development toolkit in which "[n]ouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms, each expressing a distinct concept" [4].

Words are treated (and encapsulated) as objects, with properties and relationships to other words that can be evaluated and used in realtime; it becomes possible to sonify text as whole words, using well-defined relationships between different words, rather than sonifying text as characters or keyboard events alone. However, this raw power delivers us a 'blank-slate' problem: how do we create an appropriate framework for this linguistic data? How do we incorporate these extra dimensions of words into a Human Computer Interface (HCI)? Use them as a creative tool that can be meaningfully integrated with music?

Magnusson [5] presents a useful discussion in which designing music software is framed as a semiotic act, "structur[ing] a system of signs into a coherent whole that incorporates some compositional ideology (or an effort to exclude it)". He draws a distinction between traditional HCI design (representational and task-based, often imitating real-world tasks in order to prepare and organise information) and the type of design that uses the computer for artistic creation. The distinction between the two, however, is problematic. Magnusson argues that whilst a user engaged in creative practice "deploy[s] software to achieve some end goals...this very software is also a system of representational meanings, thus influencing and coercing the artist into certain work patterns." This is as true for the most permissive musical software as it is for the most restrictive, seen in the user modifications (such as GUI extensions and sliders) created even for such flexible programs as pd.

In the case of text to music converters, software at both ends of the spectrum remains creatively restrictive despite program sophistication and flexibility. The NLTK, for example, has the ability to track words along multiple axes (synonym, homonym, antonym etc.) yet it still treats words only as raw data; music produced through a linkage to the NLTK that is not based on a structured relationship between the performer, language and sound can harness no more of the power of language than an ASCII conversion. In a creative environment increasingly rich with collaborative and multi-modal performances, there exists a gap, a loss of meaning in the translation of text to sound. Common performance techniques include: poetry performance with sonification designed and improvised in response to the performed poetry in real time or the previously mentioned ASCII conversion of typed text, perhaps with extra manipulation [6]. In many performances which use data to generate sound, the source of the data, or its potential 'meaning' is often considered largely irrelevant. However, where text is employed as the data source it is mostly on display in some form, whether auditory or visual. This would seem to imply that the text/sound relationship has a certain importance, which may not be fully realised in current processes of sound generation.

2. P[A]RA[PRA]XIS: A SEMI[ER]OTIC MACHINE

P[a]ra[pra]xis provides a platform for the performer (or musician, or writer) to sculpt a personally meaningful system of linguistic substitution within a self-created text. Although the *P[a]ra[pra]xis Suite* software is applicable to any project involving the sonification of data gathered from lingual substitutions, it was created with a particular direction in mind. The term 'Parapraxis' emerged as an English translation for what Freud termed *die Fehlleistung*, literally, 'faulty action', used to describe the unintentional miscommunication occurring during even the most banal of daily human interactions [7]. It encompasses the range of mistaken perceptions, actions or speech which occur when the subconscious and the conscious mind, as is generally the case, are working to non-aligned agendas, and is commonly known as the Freudian slip, where you may 'say one thing but mean your mother'. Needless to say, its motives are often classed as sexual.

The unique combinations of words and concepts which parapraxis creates also lend an additional flexibility to grammatical norms. Whereas Freud's 'parapraxis' is either a singular instance or a genre-descriptor of such an error and constitutes that which is a kind of 'sub-normal activity' in relation to the business of perception and communication, our version, P[a]ra[pra]xis, conflates the nuance of 'para' meaning 'beyond', or 'outside of' with the academic notion of 'praxis' as theory put into action: thus it comes to describe an entire way of creatively exploring language and music through the building of user-initiated dictionaries based on free association and metonymic slippage [8].

In the early 1900s, the Swiss linguist, Ferdinand de Saussure, was responsible for the development of a linguistic apparatus which re-defined the focus of the relationship between words and the ways in which meanings become attached to them. Saussure claimed the linguistic sign as "a two-sided psychological entity", consisting only of "a *concept* which exists in equilibrial relationship with a *sound pattern*" [9].

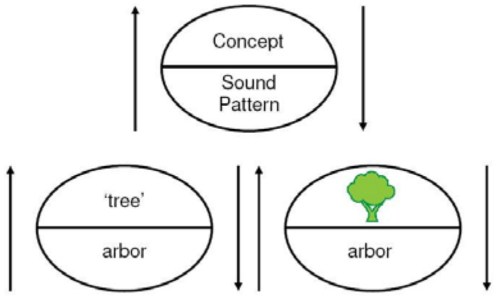


Figure 1. de Saussure's diagram, demonstrating the relationship between concept and sound pattern

In one of Saussure's most well-known diagrams (See Figure 1. above) the concept is designated by a word which 'stands in for' an actual physical object. The 'word' *tree* (or *arbor*, in Saussure's native tongue) has nothing to do with either the image it conjures up, or the physical reality of a tree. This idea, that sign and signified have no innate connection, has played out in many different guises over the course of the last hundred odd years, beginning with early modernism, and culminating in multiple instances of user-created semiotic systems, where any sign may be attached to any signifier, as long as the relationship is pre-determined. In the paper previously mentioned, Magnusson sees that "actors and the contexts in which they function are all elements in a semiotic language…We provide a semiotics or suggest language games where the behaviour of an actor maps onto some parameters in a sound engine. For example, vertical location of an actor could signify the pitch of a tone or playback rate of a sample"[5].

In taking on Saussure's notion that 'the link between signal and signification is arbitrary', many conceptual versions of semiotic systems fail to take a key factor into account: much of the power of language arises precisely *because of* the false innate meaning we ascribe to individual words. *P[a]ra[pra]xis* aims to utilise this power by involving the performer/user in a tension between emotional or psychic resonances which may be attached to particular word significations and the implementation of a rule-set which can make what may at first appear to be extremely radical changes to the associations between words as we generally use them.

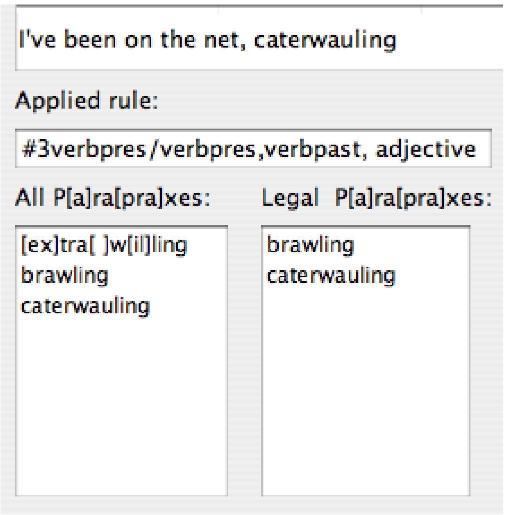
This returns us to Freud's investigation of the hidden associations lurking in every Parapraxis; *P[a]ra[pra]xis* works to open up these associations in several ways. Firstly, a user involved in entering or modifying words for the dictionary file is free to explore their own mental links between sounds, text and ideas. When dealing with the word '**box**', one man's '**bo[ra]x**' may be another man's '**b[ot]ox**'. When playing *P[a]ra[pra]xis* in real-time, users will be forced to respond to lingual substitutions determined by a dynamic, but grammatically oriented rule-set. A player writing a poem or story will be subjected to a continually altering narrative, and will thus involuntarily form new chains of signification, by either engaging or refusing to engage with the material presented.

3. THE P[A]RA[PRA]XIS SOFTWARE SUITE

The language substitutions that occur when a performer enters a dictionary word in the *P[a]ra[pra]xis* set are predicated on six linguistic conditions: anagram; phonetic substitution; predictive; additive; subtractive; midrash. For a more detailed outline, please see [10]. The *P[a]ra[pra]xis Software Suite* includes two applications which together enable the creation of and implementation of the word substitution process described above.

P[a]ra[pra]xis Collection Editor is a straightforward Java application which manages the relationships between words and their possible substitutions. *Realtime P[a]ra[pra]xis* is also a Java application; it handles the realtime implementation of rules designed within it on a dictionary file created in the Collection Editor. Here, a rule describes the conditions that must be met for a word to be substituted by another word.

A typed word is only replaced if two conditions are met: the typed word exists as an 'original' word in the dictionary; and the typed word has at least one substitution that meets the conditions of a rule. For example, if the rule stipulates that nouns can only be replaced with other nouns, and the typed word is a noun but none of its possible replacements are nouns, no substitution is made. Figure 2. shows how a set of possible substitutions are filtered into a set of legal substitutions.



I've been on the net, caterwauling

Applied rule:

#3verbpres/verbpres,verbpast, adjective

All P[a]ra[pra]xes:
[ex]tra[ ]w[il]ling
brawling
caterwauling

Legal P[a]ra[pra]xes:
brawling
caterwauling

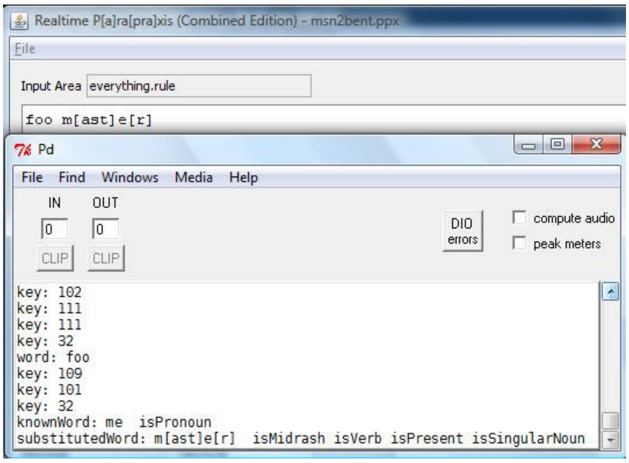
Figure 2. A screenshot from *Realtime P[a]ra[pra]xis*. A user has just typed 'I've been on the net, trawling' when 'trawling' is found to have possible word substitutions, shown in the 'All P[a]ra[pra]xes' column. The rule here, however, only allows present-tense verbs to be replaced with a word that is either an adjective or a past-/present-tense verb. Further, this rule only permits a phonetic substitution, disallowing '[ex]tra[]w[il]ling' as a possible substitution.

Realtime Parapraxis broadcasts on four different OSC address patterns which, interpreted together, give an external application insight across the continuum of a performer's input:

1) /key (Integer): the ASCII code for each key typed;
2) /word (String): a String that contains the last word typed. This is sent whenever a non-alphanumeric character is typed and the system assumes a word has been completed, but this word is not found in the original word list in the dictionary;
3) /knownWord (String): a String that contains the last word typed if that word appears in the dictionary's original word list - whether or not the word has been substituted - followed by a list containing that word's properties; and
4) /replacement (String): a String that contains the word that replaced the last word typed only if a legal substation was made, followed by a list containing that replacement word's properties.

Figure 3. shows the OSC output as received in pd.

Further, Parapraxis Collection Editor allows users to create their own word/replacement descriptors as well as the standard properties. These are appended sequentially to the list broadcast on the /knownWord and /replacement address patterns[1].


Realtime P[a]ra[pra]xis (Combined Edition) - msn2bent.ppx
File
Pd
File Find Windows Media Help
IN OUT
CLIP CLIP
DIO errors
compute audio
peak meters

Input Area everything.rule

```
foo m[ast]e[r]
```

```
key: 102
key: 111
key: 111
key: 32
word: foo
key: 109
key: 101
key: 32
knownWord: me  isPronoun
substitutedWord: m[ast]e[r]  isMidrash isVerb isPresent isSingularNoun
```

Figure 3. Pd receives information regarding the performer's input in realtime. Key presses, unknown words, known words and (when applicable) replacement words are broadcast, along with their properties.

As well as the two Java applications, we are making a Java library (JAR file) available which provides all the functionality for *P[a]ra[pra]xis*. This library can be used to develop custom graphical interfaces as well as manipulating word and word substitution relationships in a unique way and from the ground up.

4. CASE STUDY

Po[or Symm]etry [Dra]in[s] [E]motion[s] is a live networked performance piece developed with *P[a]ra[pra]xis* software and implemented using the JAR library and a custom GUI. It should be stressed that this description is not prescriptive; lyrical and musical decisions are entirely decoupled from the core software.

The work presents an Instant Messenger (IM)-like conversation between two people, in an obviously troubled relationship. Both screens are presented separately to the audience.

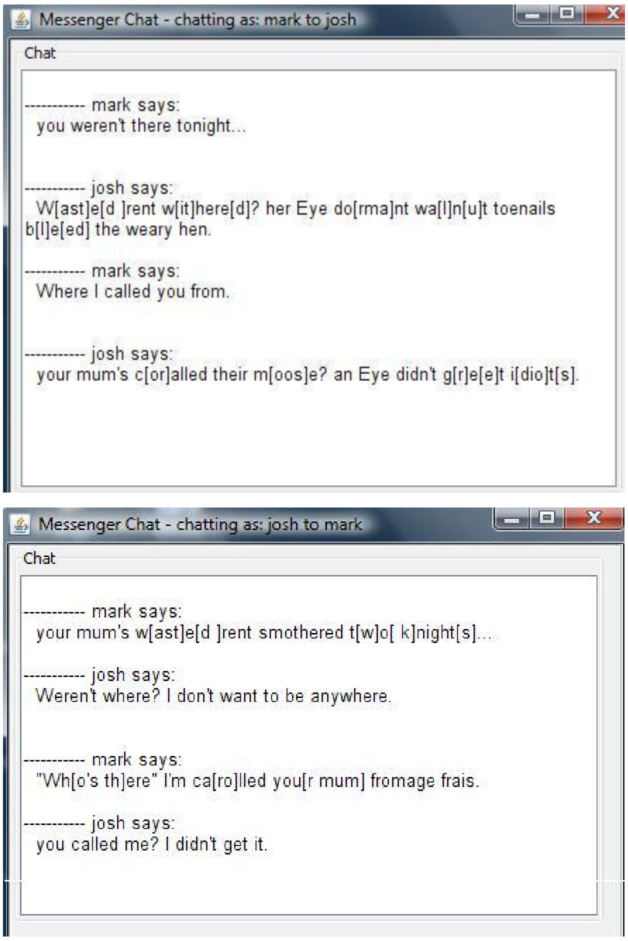
As performer A begins to type, the text is displayed unadulterated on their screen. When they click 'send', however, the text briefly appears in its original form on performer B's screen before being 're-written', converted on screen as though it were being typed. Figure 4 shows part of this conversation.

The dictionary for this piece consists of about 160 words; there are currently 370 possible substitutions which can be made. Even if the performers have worked with the piece before, there will still be linguistic surprises.

The piece presents both an auditory and a visual rendering of the ways in the 'meaning' of language shifts: from 'speaker' to

[1] Whilst the default OSC output is a space-separated string that contains a word's properties, the software can also output a list of Boolean states that may be more easily interpreted in different music software.

'hearer' and from the utilitarian meanings we ascribe to words for the sake of shared communication to the metonymic resonances (often unwelcome) which are engendered in the unconscious mind.


Messenger Chat - chatting as: mark to josh
Chat

---------- mark says:
you weren't there tonight...

---------- josh says:
W[ast]e[d]rent w[it]here[d]? her Eye do[rma]nt wa[l]n[u]t toenails b[l]e[ed] the weary hen.

---------- mark says:
Where I called you from.

---------- josh says:
your mum's c[or]alled their m[oos]e? an Eye didn't g[r]e[e]t i[dio]t[s].

Messenger Chat - chatting as: josh to mark
Chat

---------- mark says:
your mum's w[ast]e[d]rent smothered t[w]o[k]night[s]...

---------- josh says:
Weren't where? I don't want to be anywhere.

---------- mark says:
"Wh[o's th]ere" I'm ca[ro]lled you[r mum] fromage frais.

---------- josh says:
you called me? I didn't get it.

Figure 4. Screen shots of the IM performance. Each person sees the original text they type whilst only seeing the altered version of the other person's text.

Because the text is re-written in realtime on the other person's screen, (typically animated at around 25 mSec/character), the performance develops its own pace. Also, a visual counterpoint develops between the two screens, as the square brackets make substituted sections appear especially dense.

The music is generated by interpreting a number of performance artefacts. Based on a set of endless glissandi [Risset], their relative base frequencies and speed are continually modified as a counterpoint to the tension in the dialogue. Specific factors controlling musical parameters are: average time between keystrokes; sentence length; phrase length (how much a person types before pressing the 'send' button); and type of substitution. Interpreting the type of substitution is especially powerful. Whilst most of the substitutions are midrashes and use square brackets, phonetic substitutions and anagrams provide visual relief as well as prompting a different kind of intellectual reaction from an audience. The music-generating algorithm uses these to structure the relationships between glissandi in a fugal counterpoint, and signaling the start of a new invocation of the cantus firmus, or principle melodic line.

As the performer/musician/writer has complete control not only over the possible substitutions created for dictionary words, but also over the framework in which to define their relationships, it is very easy to generate audio output which maps the emotionality of the piece through changes in the text.

5. CONCLUSION

The development of this *P[a]ra[pra]xis* software suite marks a milestone in a continually evolving and expanding project. Starting from the simple shared idea of a basic real-time interactive poetry generator, we have been drawn to grammar, linguistics, psychoanalytical theory and serial, electronic composition as tools to investigate the human relationship to language.

P[a]ra[pra]xis marks a collaboration between two authors from divergent backgrounds within the Creative Arts field; Poetry and Sonic Arts. In order to make P[a]ra[pra]xis a genuine collaboration, not just an outsourcing of difficult specialist tasks, we have had to adjust and develop our perceptions of our own and each others' language, just as those who play P[a]ra[pra]xis will. Hopefully others will find this as beneficial as we have.

6. REFERENCES

[1]Judge,A.<http://www.laetusinpraesens.org/docs00s/convert.php> (draft, 2007)

[2] Worral, D., Bylstra, M., Barrass, S., and Dean, R. *Sonipy: The Design of an Extendable Software Framework for Sonification Research and Auditory Display* Proc. ICAD 2007, Montreal, Canada.

[3] NLTK <http://nltk.sourceforge.net >

[4] Wordnet <http://wordnet.princeton.edu>

[5] Magnusson, T. *Screen-Based Musical Interfaces as Semiotic Machines* Proc. NIME 2006, Paris, France.

[6] See, for example, the work of Jenkins, G.S. at <http://www.1-4inch.com/archive05.html>

[7] Freud, S. 'The Psychopathology of Everyday Life' in *The Standard Edition of the Complete Psychological Works of Sigmund Freud*, Vol. VI (London: The Hogarth Press and the Institute of Psychoanalysis, 1966).

[8] Detailed discussion on the role played by metonymic slippage in the functioning of the unconscious can be found in: Lacan, J. *The Four Fundamental Concepts of Psycho-analysis*, ed. Jacques-Alain Miller, trans. Alan Sheridan (London: Hogarth Press, 1973)

[9] de Saussure, F. *Course in general linguistics*, ed. Charles Bally and Albert Sechehaye with Albert Riedlinger, trans. Roy Harris, (London: Duckworth, 1983), 67.

[10] Dubrau, J. and Havryliv, M. *P[a]ra[pra]xis* in proc. ACMC, June 2007