Neural, Parallel, and Scientific Computations 20 (2012) 227-240

PULSED MELODIC AFFECTIVE PROCESSING - USING MUSIC FOR NATURAL AFFECTIVE COMPUTATION AND INCREASED PROCESSING TRANSPARENCY

ALEXIS KIRKE AND EDUARDO MIRANDA

University of Plymouth, Drake Circus, Plymouth, PL4 8AA, UK
Emails: {Alexis.kirke, Eduardo.miranda}@plymouth.ac.uk

ABSTRACT. Pulsed Melodic Affective Processing (PMAP) is a computation protocol useable at ultiple levels in data processing systems. The approach utilizes musically-based pulse sets ("melodies") for processing – capable of representing the arousal and valence of affective states. Affective processing and affective input/output is now considered to be a key tool in artificial intelligence and computing. In the designing of processing elements (e.g. bits, bytes, floats, etc), engineers have primarily focused on the processing efficiency and power. Having defined these elements, they then go on to investigate ways of making them perceivable by the user/engineer. However the extremely active and productive area of Human-Computer Interaction - and the increasing complexity and pervasiveness of computation in our daily lives – supports the idea of a complementary approach in which computational efficiency and power are more balanced with understandability to the user/engineer. PMAP provides the potential for a person to tap into the affective processing path to hear a sample of what is going on in that computation, as well as providing a simpler way to interface with affective input/output systems. This comes at a cost of developing new approaches to processing and interfacing PMAP-based modules - this cost being part of the compromise of efficiency/power versus user-transparency and interfacing. In this paper we introduce and develop PMAP; and demonstrate and examine the approach using two example applications: a military robot team simulation with an affective subsystem, and a text affective-content estimation system.

Key Words: HCI, Logic, Neural Networks, Affective Computing, Fuzzy Logic, Computer Music.

1. INTRODUCTION

This paper proposes the use of music as a processing tool for affective computation in artificial systems. It has been shown that affective states (emotions) play a vital role in human cognitive processing and expression [1]. As a result affective state processing has been incorporated into artificial intelligence processing and robotics [2]. The issue of developing systems with affective intelligence which also provide for greater user-transparency, is what is addressed in this paper.

Music has often been described as a language of emotions [3]. The general features which express emotion in western music are known. Before introducing these, affective representation will be discussed. The dimensional approach to specifying emotion utilizes an n-dimensional space made up of emotion "factors". Any emotion can be plotted as some combination of these factors. For example, in many emotional music systems [4] two dimensions are used: Valence and Arousal (see Figure 1). In that model, emotions are plotted on a graph with the first dimension being how positive or negative the emotion is (Valence), and the second dimension being how intense the physical arousal of the emotion is (Arousal). For example "Happy" is high valence high arousal affective state, and "Stressed" is low valence high arousal state.

Received September 15, 2011

Previous research [5] has suggested that a main indicator of valence is musical key mode: major or minor. An example of a minor key piece of music is Beethoven's Moonlight Sonata. An example of a major key piece of music is the Spring movement of Vivaldi's Four Seasons. A Major key implies higher valence, minor key implies lower valence. It has been shown that tempo is a prime indicator of arousal. High tempo indicating higher arousal, low tempo - low arousal. There has been work into automated systems which communicate emotions through music [6] and which detect emotion embedded in music based on musical features [7].

Affective Computing [8] focuses on robot/computer affective *input/output*. Whereas a key aim of PMAP is to develop data streams that represent such affective states, and use these representations to process data and compute actions. The other aim of PMAP is more related to Picard's work – to aid easier sonification of affective processing [9] for user transparency, i.e. representing non-musical data in musical form to aid its understanding. Related sonification research has included tools for using music to debug programs [10].

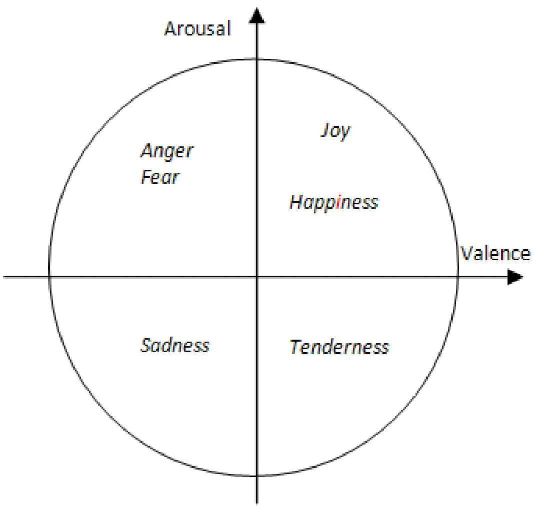

Figure 1: The Valence/Arousal Model of Emotion

2. PMAP REPRESENTATION OF AFFECTIVE STATE

Pulsed melodic affective processing (PMAP) is a method of representing affective state using music. In PMAP the data stream representing affective state is a stream of pulses of 10 possible different levels. For example 10 different voltage levels for a low level stream, or 10 different integer values for a stream embedded in some sort of data structure. Each level represents one of the musical notes *C,D,Eb,E,F,G,Ab,A,Bb,B*. For example 1mV could be C, 2mV be D, 3mV be Eb, etc. We will simply use integers here to represent the notes (i.e. 1 for C, 2 for D, 3 for Eb, etc). These note values are designed to represent a valence (positivity or negativity of emotion). This is because, in the key of C, pulse streams made up of only the notes *C,D,E,F,G,A,B* are the notes of the key C major, and so will be heard as having a major key mode – i.e. positive valence. Whereas streams made up of *C,D,Eb,F,G,Ab,Bb* are the notes of the key C minor, and so will be heard as having a minor key mode – i.e. negative valence. Furthermore, the pulses are transmitted at a variable rate. This pulse rate represents the arousal, since a higher pulse rate is essentially a series of pitches played at a high tempo (high arousal); whereas a lower pulse rate is a series of pitches played at a low tempo (low arousal).

For example a PMAP stream of say [C, C, Eb, F, D, Eb, F, G, Ab, C] (i.e. [1,1,3,5,3,4,5,6,7]) would be principally negative valence because it is mainly minor key mode. Whereas [C,C,E,F,D,E,F,G,A,C] (i.e. [1,1,4,5,2,4,5,6,8]) would be seen as principally positive valence. And the arousal of the pulse stream would be encoded in the rate at which the pulses were transmitted. So if [1,1,3,5,3,4,5,6,7] was transmitted at a high rate, it would be high arousal and high valence – i.e. a stream representing ‘happy’. Where as if [1,1,4,5,2,4,5,6,8] was transmitted at a low pulse rate then it will be low arousal and low valence – i.e. a stream representing ‘sad’.

Note that [1,1,3,5,3,4,5,6,7] and [3,1,3,5,1,7,6,4,5] both represent high valence (i.e. are both major key melodies in C). This has a potential extra use. If there are two modules or elements both with the same affective state, the different note groups which make up that state representation can be unique to the object generating them. This allows other objects, and human listeners, to identify where the affective data is coming from.

In performing some of the initial analysis on PMAP, it is convenient to utilize a parametric form to represent the data stream form. The parametric form represents a stream by a Tempo-value variable and a Key-mode-value variable. The Tempo-value is a real number varying between 0 (minimum pulse rate) and 1 (maximum pulse rate). The Key-mode-value is an integer varying between -3 (maximally minor) and 3 (maximally major).

3. MUSICAL LOGIC GATES

Looking now at elements which process PMAP data streams, three gates which take PMAP data as inputs will be examined based on AND, OR and NOT logic gates. The PMAP versions of these are respectively: MNOT (pronounced “emm-not”), MAND, and MOR. The normal logic gates take binary streams at a constant data rate as input, whereas PMAP has more than two levels and is variable rate. A given PMAP stream can be represented by a PMAP-value which can be written as $m_i = [k_i, t_i]$ with key-mode-value k_i and tempo-value t_i. m_i is also referred to as the “KT-Value” of the stream. The definitions of the musical gates are (for two PMAP streams m_1 and m_2):

$$MNOT(m) = [-k, 1-t] \quad (1)$$

$$m1 \; MAND \; m2 = [minimum(k_1,k_2), minimum(t_1,t_2)] \quad (2)$$

$$m1 \; MOR \; m2 = [maximum(k_1,k_2), maximum(t_1,t_2)] \quad (3)$$

These use a similar approach to Fuzzy Logic [11]. MNOT is the simplest – it simply reverses the key mode and tempo – minor becomes major and fast becomes slow, and vice versa. The best way to get some insight into what the affective function of the music gates is it to utilize music “truth tables”, which will be called Affect Tables here. In these, four representative state-labels are used to represent the four quadrants of the PMAP-value table: “Sad” for [-3,0], “Stressed” for [-3,1], “Relaxed” for [3,0], and “Happy” for [3,1]. Table 1 shows the music tables for MNOT and MAND.

Table 1: *Music Tables for MAND and MNOT*

MAND						**MNOT**			
State Label 1	*State Label 2*	*KT-value 1*	*KT-value 2*	*MAND value*	*State Label*	*State Label*	*KT-value*	*MNOT value*	*State Label*
Sad	Sad	-3,0	-3,0	-3,0	Sad	Sad	-3,0	3,1	Happy
Sad	Stressed	-3,0	-3,1	-3,0	Sad	Stressed	-3,1	3,0	Relaxed
Sad	Relaxed	-3,0	3,0	-3,0	Sad	Relaxed	3,0	-3,1	Stressed
Sad	Happy	-3,0	3,1	-3,0	Sad	Happy	3,1	-3,0	Sad
Stressed	Stressed	-3,1	-3,1	-3,1	Stressed				
Stressed	Relaxed	-3,1	3,0	-3,0	Sad				
Stressed	Happy	-3,1	3,1	-3,1	Stressed				
Relaxed	Relaxed	3,0	3,0	3,0	Relaxed				
Relaxed	Happy	3,0	3,1	3,0	Relaxed				
Happy	Happy	3,1	3,1	3,1	Happy				

Taking the MAND of two melodies, low tempos and minor key modes will dominate over high tempos and major key modes. Taking the MOR of two melodies, then the high tempos and major keys will dominate the output. Another way of viewing this is that the MAND of the melodies from Moonlight Sonata (minor key, low tempo) and the Marriage of Figaro Overture (major key, high tempo), the result would be mainly influenced by Moonlight Sonata. However if they are MOR'd, then the Marriage of Figaro Overture would dominate. The MNOT of Marriage of Figaro Overture would be a slow minor key version. The MNOT of Moonlight Sonata would be a faster major key version. It is also possible to construct more complex music functions. For example MXOR (pronounced "mex-or"):

$$m_1 \text{ MXOR } m_2 = (m_1 \text{ MAND MNOT}(m_2)) \text{ MOR } (\text{MNOT}(m_1) \text{ MAND } m_2) \quad (5)$$

A simple application is now examined. One function of affective states in biological systems is that they provide a back-up for when the organism is damaged or in more extreme states [12]. For example an injured person who cannot think clearly, will still try to get to safety or shelter. An affective subsystem for a robot who is a member of a military team is now examined; one that can kick in or over-ride if the higher cognition functions are damaged or deadlocked. Figure 2 shows the system diagram. A group of mobile robots with built-in weapons are placed in a potentially hostile environment and required to search the environment for enemies; and upon finding enemies to move towards them and fire on them. The PMAP affective sub-system in Figure 2 is designed to keep friendly robots apart (so as to maximize the coverage of the space), to make them move towards enemies, and to make them fire when enemies are detected.

The modules in Figure 2 are "DetectOther", "FriendFlag", "MOTOR", and "WEAPON". "DetectOther" emits a regular minor melody; then every time another agent (human or robot) is detected within firing range, a major-key melody is emitted. This is because detecting another agent means that the robots are not spread out enough if it is a friendly, or it is an enemy if not. "FriendFlag" emits a regular minor key melody except for one condition. Other friends are identifiable (visually or by RFI) - when an agent is detected within range, and if it is a friendly robot – this module emits a major key melody. "MOTOR" – this unit, when it receives a C major key note (*E*, *A*, or *B*) moves the robot forward one step. When it receives a C minor key note (*Eb*, *Ab*, or *Bb*) it moves the robot back one step. "WEAPON" - this unit, when it receives a minor key note fires one round. The weapon and motor system is written symbolically in equations (4) and (5):

$$WEAPON = DetectOther\ MAND\ MNOT(FriendFlag) \qquad (4)$$
$$MOTOR = WEAPON\ MOR\ MNOT(DetectOther) \qquad (5)$$

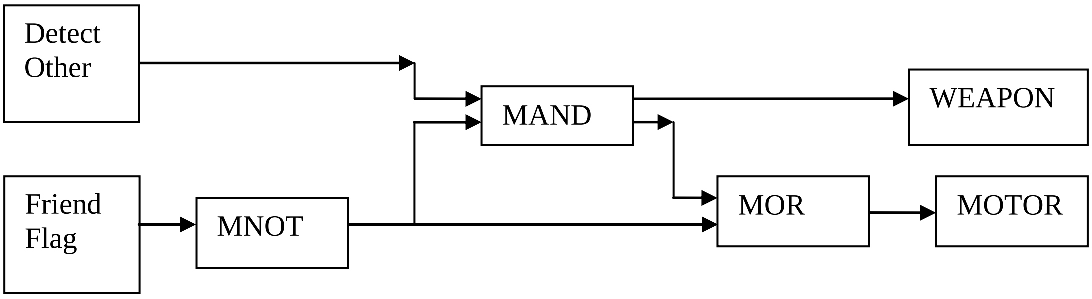

Figure 2: Affective Subsystem for Military Multi-robot System

Using Equations (1) and (2) gives the theoretical results in Table 2. The 5 rows displayed are the only feasible input state combinations, and have the following interpretations respectively: (a) If alone continue to patrol and explore; (b) If a distant enemy is detected move towards it fast and start firing slowly; (c) If a distant friendly robot is detected move away so as to patrol a different area of the space; (d) If enemy is close-by move slowly (to stay in its vicinity) and fire fast; (e) If a close friend is detected move away. This should mainly happen (because of row c) when robot team are initially deployed and they are bunched together, hence slow movement to prevent collision.

Table 2: *Theoretical Effects of Affective Subsystem*

Detect Other	***Friend Flag***	***Detect Other-Value***	***Friend Flag-Value***	***MNOT (Friend Flag)***	***MAND Detect Other***	***WEAPON***	***MNOT (Detect Other)***	***MOR WEAPON***	***MOTOR***
Sad	Sad	-3,0	-3,0	3,1	-3,0	inactive	3,1	3,1	Fast forwards
Relaxed	Sad	3,0	-3,0	3,1	3,0	Firing	-3,1	3,1	Fast forwards
Relaxed	Relaxed	3,0	3,0	-3,1	-3,0	Inactive	-3,1	-3,0	Slow back
Happy	Stressed	3,1	-3,1	3,0	3,0	Firing	-3,0	3,0	Slow forwards
Happy	Happy	3,1	3,1	-3,0	-3,0	inactive	-3,0	-3,0	Slow back

To test in simulation, four friendly robots are used, implementing the PMAP-value processing described earlier, rather than having actual melodies within the processing system. The robots using the PMAP affective sub-system are called “F-Robots” (friendly robots). The movement space is limited by a border and when an F-Robot hits this border, it moves back a step and tries another movement. Their movements include a perturbation system which adds a random nudge to the robot movement, on top of the affectively-controlled movement described earlier. The simulation space of is 50 units by 50 units. An F-Robot can move by up to 8 units at a time backwards or forwards. Its range (for firing and for detection by others) is 10 units. Its PMAP minimum tempo is 100 beats per minute (BPM), and its maximum is 200 BPM. These are encoded as a tempo value of 0.5 and 1 respectively. Stationary enemy sentry robots are placed at fixed positions (10,10), (20,20) and (30,30).

The F-robots are placed at initial positions (10,5), (20,5), (30,5), (40,5), (50,5)– i.e. they start at the bottom of the space. The system is run for 2000 movement cycles – in each movement cycle each of the 4 F-Robots can move. 30 simulations were run and the average distance of the F-Robots to the enemy robots was calculated. Also the average distances between F-Robots was calculated. These were done with a range of 10 and a range of 0. A range of 0 effectively switches off the musical processing. Hence the results shown in Table 3 refer to PMAP system switched off and PMAP switched on respectively. It can be seen that the affective subsystem keeps the F-Robots apart encouraging them to search different parts of the space. In fact it increases the average distance between them by 72%. Similarly the music logic system increases the likelihood of the F-Robots moving towards enemy robots. The average distance between the F-Robots and the enemies decreases by 21% thanks to the melodic subsystem. And these results are fairly robust with coefficients of variation between 4% and 2% respectively across the results.

Table 3: *Results for Robot Affective Subsystem*

Range	**Avg Distance between F-Robots**	**Std Deviation**	**Average Distance of F-Robots from Enemy**	**Std Deviation**
0	7.6	0.5	30.4	0.3
10	13.1	0.5	25.2	0.4

It was also found that the WEAPON firing rate had a very strong tendency to be higher as enemies were closer. This is shown in Figure 3. The x-axis is distance from the closest enemy, and the y-axis is tempo. It can be seen that the maximum tempo (just under maximum tempo 1) or firing rate is achieved when the distance is at its minimum. Similarly the minimum firing rate occurs at distance 10 in most cases. In fact the correlation between the two was found to be -0.98 which is very high. The line is not straight and uniform because it is possible for robot 1 to be affected by its distance from other enemies and from other friendly robots.

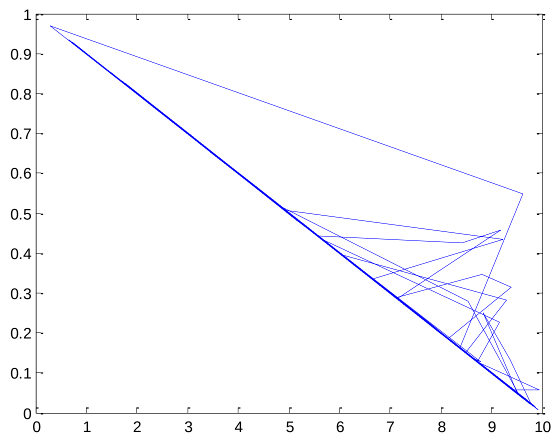

Figure 3: Plot of distance of robot 1 from enemy (when firing) and its weapons' tempo value

Finally it is worth considering what these robots actually sound like as they move and change status. To allow this each of the 4 robots was assigned a distinctive motif, with constant tempo. Motives designed to identify a module, agent, etc will be called "Indentive". The identives for the 4 robots were:

1. [1,2,3,5,3,2,1] = C,D,Eb,F,Eb,D,C
2. [3,5,6,7,6,5,3] = Eb,F,G,Ab,G,F,Eb
3. [6,7,9,1,9,7,6] = G,Ab,Bb,C,Bb,Ab,G
4. [7,9,1,6,1,9,7] = Ab,Bb,C,G,C,Bb,Ab

Figure 4 shows the first 500 notes of robots 1 to 3 in the simulation in piano roll notation. The octave separation used for the Figure 4 also helped with aural perception. (So this points towards octave independence in processing as being a useful feature.) It was found that more than 3 robots was not really perceivable. It was also found that transforming the tempo minimums and maximums to between 100 and 200 beats per minute and quantizing by 0.25 beats seemed to make seem to make changes more perceivable as well. In this way a human commander can monitor the state of a robot team in parallel based on their consonance, dissonance and synchronization.

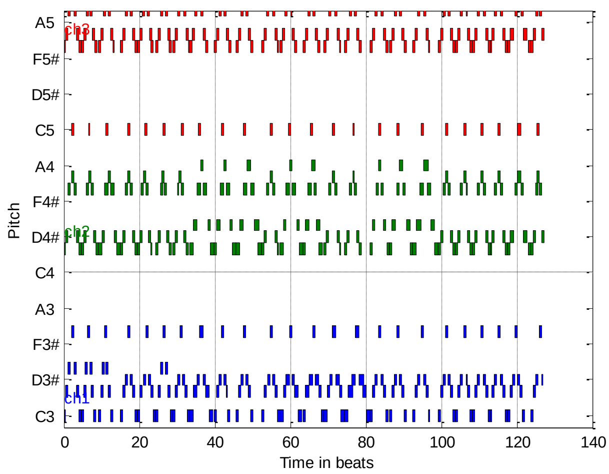

Figure 4: A plot of 500 notes in the “motor” processing of robots 1 to 3 (octave separated).

An extension of this system is to incorporate rhythmic biosignals from modern military suits [13][14]. For example if “BioSignal” is a tune generating module whose tempo is a heart rate reading from a human soldiers military body suit, and whose key is based on EEG valence readings, then the MOTOR system becomes:

$$MOTOR = WEAPON\ MOR\ MNOT(DetectOther) \qquad MOR\ MNOT(BioSignal) \qquad (6)$$

The diagram for (6) is shown in Figure 5, and the music table in Table 4. The table shows that if a (human) friend is detected whose biosignal indicates positive valence, then the F-robot will move away from the friend to patrol a different area. If the friendly human’s biosignal is negative then the robot will move towards them to aid them.

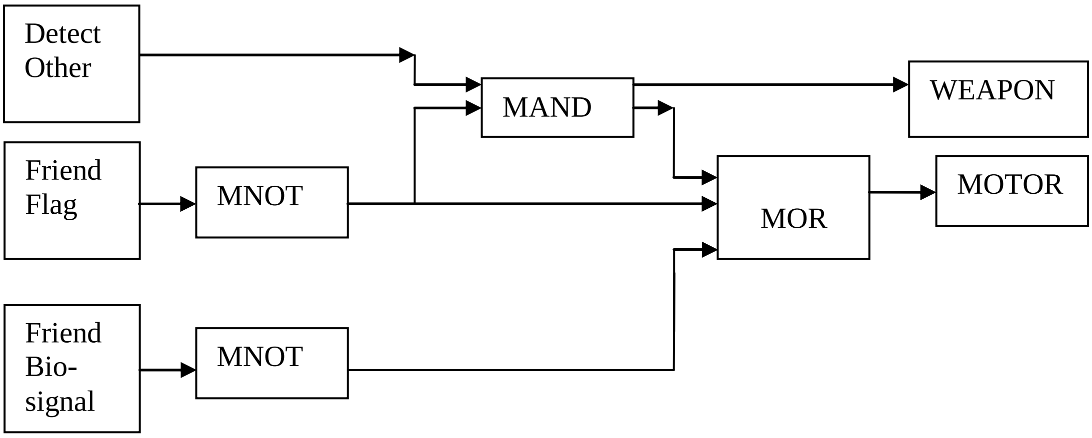

Figure 5: *Affective Subsystem incorporating Biosignals*

4. MUSICAL NEURAL NETWORKS

We will now look at a form of learning artificial neural network which uses PMAP. These artificial networks take as input, and use as their processing data, pulsed melodies. A musical neuron (muron – pronounced MEW-RON) is shown in Figure 6. The muron in this example has two inputs, though it can have more than this. Each input is a PMAP melody, and the output is a PMAP melody. The weights on the input w_1 and w_2 are two element vectors which define a key transposition, and a tempo change. A positive R_k will make the input tune more major by selective transposition, and a negative one will make it more minor. (Transposition is the process of switching notes unique to one key mode into another, e.g. *E* to *Eb* or *Ab* to *A*. The more of these notes that are switched between, the greater will be the perception of changing key mode.) A positive D_t will increase the tempo of the tune, and a negative D_t will reduce the tempo. The muron combines input tunes by superposing the spikes in time – i.e. overlaying them. Any notes which occur at the same time are combined into a single note with the highest pitch being retained. Murons can be combined into networks, called musical neural networks, abbreviated to "MNN". The learning of a muron involves setting the weights to give the desired output tunes for the given input tunes. Applications for which PMAP is most efficiently used are those that naturally utilize temporal or affective data (or for which internal or external sonification is particularly important).

Table 4: *Music table for Biosignal Extension*

Detect Other	***Friend Flag***	***Bio-signal***	***Bio-signal Value***	***MNOT (Bio-Signal)***	***MNOT (Friend Flag)***	***WEAPON***	***MOR WEAPON***	***MOR MNOT (Bio-Signal)***	***MOTOR***
Relaxed	Relaxed	Sad	-3,0	3,1	-3,1	Inactive	-3,0	3,1	forwards
Relaxed	Relaxed	Stressed	-3,1	3,0	-3,1	Inactive	-3,0	3,0	forwards
Relaxed	Relaxed	Relaxed	3,0	-3,1	-3,1	Inactive	-3,0	-3,1	backwards
Relaxed	Relaxed	Happy	3,1	-3,0	-3,1	Inactive	-3,0	-3,0	backwards
Happy	Happy	Sad	-3,0	3,1	-3,0	inactive	-3,0	3,1	forwards
Happy	Happy	Stressed	-3,1	3,0	-3,0	inactive	-3,0	3,0	forwards
Happy	Happy	Relaxed	3,0	-3,1	-3,0	inactive	-3,0	-3,1	backwards
Happy	Happy	Happy	3,1	-3,0	-3,0	inactive	-3,0	-3,0	backwards

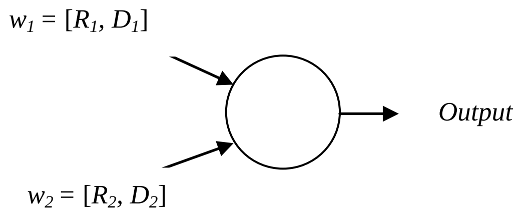

Figure 6: *A Muron with two inputs*

One such system will now be proposed for the estimation of affective content of real-time typing. The system is inspired by research by the authors on analysing QWERTY keyboard typing, in a similar way that piano keyboard playing is analyzed to estimate the emotional communication of the piano player [15]. In this a real-time system was developed to analyse tempo of typing and estimate affective state. The MNN/PMAP version demonstrated in this paper is not real-time, and does not take into account base typing speed. This is to simplify simulation and experiments here. The proposed architecture for offline text emotion estimation is shown in Figure 7. It has 2 layers known as the Input and Output layers. The input layer has four murons – which generate notes. Every time a Space character is detected, then a note is output by the Space Flag. If a comma is detected then a note is output by the comma flag, if a full stop/period then the Period Flag generates a note, and if an end of paragraph is detected then a note is output by the Paragraph flag. The idea of these 4 inputs is they represent 4 levels of the timing hierarchy in language, and we are proposing that the timing hierarchy in language has a comparable affective function to the timing hierarchy in music.

The lowest level in the linguistic hierarchy is letters, whose rate is not measured in the demo, because offline pre-typed data is used. These letters make up words (which are usually separated by a space). The words make phrases (which are often separated by commas). Phrases make up sentences (separated by full stops), and sentences make up paragraphs (separated by a paragraph end). So the tempo of the tunes output from these 4 murons represent the relative word-rate, phrase-rate, sentence-rate and paragraph rate of the typist. (Note that for data from a messenger application, the paragraph rate will represent the rate at which messages are sent). It has been found by researchers that the emotion a musical performer is trying to communicate effects not only their basic playing rate, but also the structure of the musical timing hierarchy of their performance [16]. Similarly we propose that a person's mood will affect not only their typing rate [15], but also their relative word rate and paragraph rate, and so forth. This network is based on this idea, taking as input the relative rates (the timing hierarchy) and outputing an estimate of affective communication in the text.

The input identives are built from a series of simple rising melodies. The precise shapes of the melodies are not so important here, as they are being used a carriers of affective information. The desired output of the MNN will be a tune which represents the affective estimate of the text content. A happy tune means the text structure is happy, sad means the text is sad. Normally Neural Networks are trained using a number of methods, most commonly some variation of gradient descent. A gradient descent algorithm will be used here. w_1, w_2, w_3, w_4 are all initialised to [0,1] = [Key sub-weight, Tempo sub-weight]. So initially the weights have no effect on the key, and multiply tempo by 1 – i.e. no effect. The final learned weights are also shown in Figure 6. Note, in this simulation actual tunes are used (rather than PMAP-value parameterization used in the robot simulation). In fact the Matlab MIDI toolbox is used. The documents in the training set were selected from the internet and were posted personal or news stories which were clearly summarised as sad or happy stories. 15 sad and 15 happy stories were sampled. The happy and sad tunes are defined respectively as the targets: a tempo of 90 BPM and a major key, and a tempo of 30 BPM and a minor key.

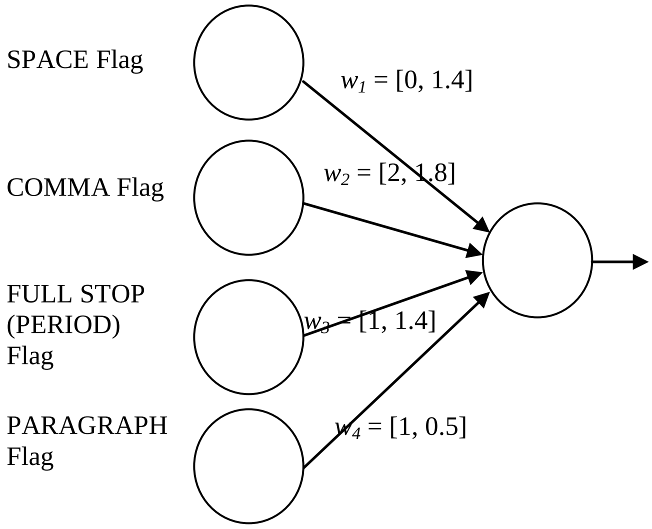

Figure 7: *MNN for Offline Text Affective Analysis*

At each step the learning algorithm selects a training document. Then it selects one of w_1, w_2, w_3, or w_4. Then the algorithm selects either the key or the tempo sub-weight. It then performs a single one-step gradient descent based on whether the document is defined as Happy or Sad (and thus whether the required output tune is meant to be Happy or Sad). The size of the one step is defined by a learning rate, separately for tempo and for key. Before training, the initial average error rate across the 30 documents was calculated. The key was measured using a modified key finding algorithm [17] which gave a value of 3 for maximally major and -3 for maximally minor. The tempo was measured in Beats per minute. The initial average error was 3.4 for key, and 30 for tempo.

After the 1920 iterations of learning the average errors reduced to 1.2 for key, and 14.1 for tempo. These results are described more specifically in Table 5 split by valence - happy or sad. Note that these are in-sample errors for a small population of 30 documents. However what is interesting is that there is clearly a significant error reduction due to gradient descent. *This shows that it is possible to fit the parameters of a musical combination unit (a muron) so as to combine musical inputs and give an affectively representative musical output, and address a non-musical problem.* (Though this system could be embedded as music into messenger software to give the user affective indications through sound). It can be seen in Table 5 that the mean tempo error for Happy documents (target 90 BPM) is 28.2 BPM. This is due to an issue similar to linear non-separability in normal artificial neural networks [18]. The Muron is approximately adding tempos linearly. So when it tries to approximate two tempos then it focuses on one more than the other – in this case the Sad tempo. Hence adding a hidden layer of murons may well help to increase reduce the Happy error significantly (though requiring some form of melodic Back Propagation).

Table 5: *Mean Error of MNN after 1920 iterations of gradient descent*

	Key Target	Mean Key Error	Tempo Target (BPM)	Mean Tempo Error (BPM)
Happy Docs	3	0.8	90	28.2
Sad Docs	-3	1.6	30	0

There is a relationship between Musical Neural Networks (MNNs) and spiking neural networks (SNNs) [19]. SNNs have been studied both as artificial entities and as part of biological neural networks in the brain. In SNN's the spike height is usually not relevant (only the spike rate or "tempo"). In MNNs the pulse ("spike") height encodes a pitch (and through that a key).We mentioned above the desirability of a back propagation algorithm for MNNs; [20] develops a back-propagation algorithm for SNNs. There has in fact been some biological work suggesting spike height, in addition to spike rate, may encode information in SNNs in the brain [21][22].

5. CONCLUSIONS

This paper has introduced the concept of pulsed melodic affective processing, a complementary approach in which computational efficiency and power are more balanced with understandability to humans (HCI); and which can naturally address rhythmic and affective processing. As examples music gates and murons have been introduced; as well as potential applications for this technology in robotics, and real-time text analysis. This paper is a summary of the research done, leaving out much of the detail and other application ideas; these include sonification experiments, ideas for implementing PMAP in a high level language, and programming by music, etc. However it demonstrates that music can be used to process affective functions either in a fixed way or via learning algorithms. The tasks are not the most efficient or accurate solutions, but have been a proof of concept of a sound-based unified approach addressing HCI and processing.

There are a significant number of issues to be further addressed with PMAP, a key one being is the rebalance between efficiency and understanding useful and practical, and also just how practical is sonification - can sonification more advanced than Geiger counters, heart rate monitors, etc really be useful and adopted? The valence/arousal coding provides simplicity, but is it sufficiently expressive while remaining simple? Similarly it needs to be considered if a different representation than tempo/key mode be better for processing or transparency. PMAP also has a close relationship to Fuzzy Logic and Spiking Neural Networks – so perhaps it can adapted based on lessons learned in these disciplines. And finally, most low level processing is done in integrated hardware. So the issues involved in designing PMAP hardware can be compared to the advantages of implementing PMAP in virtual processing systems (such virtual systems becoming more common as computing power continues to increase).

REFERENCES

[1] Malatesa, L., Karpouzis, K., Raouzaiou, A. (2009) Affective intelligence: the human face of AI, In Artificial intelligence, Springer-Verlag Berlin, Heidelberg

[2] Banik, S., Watanabe, K., Habib, M., Izumi, K. (2008) *Affection Based Multi-robot Team Work*. In Lecture Notes in Electrical Engineering, Volume 21. Springer, Berlin

[3] Cooke, D. (1959) *The Language of Music.* Oxford University Press, Oxford

[4] Kirke, A., Miranda, E. (2010) A Survey of Computer Systems for Expressive Music Performance. *ACM Surveys*, 42, 1-41.

[5] Juslin, P.N. (2005) From mimesis to catharsis: expression, perception and induction of emotion in music
D. Miell, R. MacDonald, D.J. Hargreaves (Eds.), Musical communication, Oxford University Press, Oxford, 85–115

[6] Livingstone, S.R., Muhlberger, R., Brown, A.R., Loch, A. (2007) *Controlling Musical Emotionality: An Affective Computational Architecture for Influencing Musical Emotions.* Digital Creativity, 18.

[7] Kirke, A., Miranda, E. (2011) Emergent construction of melodic pitch and hierarchy through agents communicating emotion without melodic intelligence. *International Computer Music Conference*, Huddersfield, UK, August 2011 (Accepted). International Computer Music Association.

[8] Picard, R. (2003) Affective Computing: Challenges, *International Journal of Human-Computer Studies*, 59, 55-64

[9] Cohen, J. (1994) Monitoring Background Activities. In *Auditory Display: Sonification, Audification, and Auditory Interfaces.* Addison-Wesley, MA, USA

[10] Vickers, P., Alty, J. (2003) Siren songs and swan songs debugging with music. *Communications of the ACM*, 46(7)

[11] Marinos, P. (1969) Fuzzy Logic and Its Application to Switching Systems, IEEE transactions on computers, C-18(4)

[12] Cosmides, L., Tooby, J. (2000) Evolutionary Psychology and the Emotions. In Lewis, M., Haviland-Jones, J.M. (eds), *Handbook of Emotions*. Guilford, NY.

[13] Stanford, V. (2004) Biosignals Offer Potential for Direct Interfaces and Health Monitoring, *Pervasive Computing*, 3, 99-103

[14] Kotchetkov, I., Hwang, B., Appelboom, G., Kellner, C., Sander Connolly, E. (2010) Brain-computer Interfaces: Military, Neurosurgical, and Ethical Perspective, *Neurosurgical Focus* 28(5)

[15] Kirke, A., Bonnot, M., Miranda, E. (2011) Towards using expressive performance algorithms for typist emotion detection. *International Computer Music Conference*, Huddersfield, UK, August 2011 (Accepted). International Computer Music Association.

[16] Bresin, R., Friberg, A. (2002) Emotional Coloring of Computer-Controlled Music Performances. *Computer Music Journal*, 24, 44-63.

[17] Krumhansl, C., Kessler, E. (1982) Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys, *Psychological Rev* 89, 334-368

[18] Haykin, S. (1994) *Neural Networks: A Comprehensive Foundation.* Prentice Hall, New Jersey.

[19] Natschläger, T., Ruf, B., (1998) Spatial and temporal pattern analysis via spiking neurons. Network: Computation in Neural Systems, 1998, Vol. 9, No. 3 : 319-332

[20] Bohte, S., Kok, J., La Poutre, H. (1997) Spike-prop: Error-backprogation in multi-layer networks of spiking neurons. Neurocomputing, 48(1-4), 17-37

[21] Perrinet, L., Samuelides, M., Thorpe, S. (2004) Coding static natural images using spiking event times: do neurons cooperate? IEEE Trans Neural Networks 15 1164-75

[22] Chen N, Yu J, Qian H, Ge R, Wang J-H (2010) Axons Amplify Somatic Incomplete Spikes into Uniform Amplitudes in Mouse Cortical Pyramidal Neurons. PLoS ONE 5(7): e11868