# Exploring Continuous Time Recurrent Neural Networks through Novelty Search

Steffan Ianigro
UNSW Art and Design
Cnr Oxford St and Greens Rd
Paddington NSW 2021 Australia
steffanianigro@gmail.com

Oliver Bown
UNSW Art and Design
Cnr Oxford St and Greens Rd
Paddington NSW 2021 Australia
o.bown@unsw.edu.au

## ABSTRACT

In this paper we expand on prior research into the use of Continuous Time Recurrent Neural Networks (CTRNNs) as evolvable generators of musical structures such as audio waveforms. This type of neural network has a compact structure and is capable of producing a large range of temporal dynamics. Due to these properties, we believe that CTRNNs combined with evolutionary algorithms (EA) could offer musicians many creative possibilities for the exploration of sound. In prior work, we have explored the use of interactive and target-based EA designs to tap into the creative possibilities of CTRNNs. Our results have shown promise for the use of CTRNNs in the audio domain. However, we feel neither EA designs allow both open-ended discovery and effective navigation of the CTRNN audio search space by musicians. Within this paper, we explore the possibility of using novelty search as an alternative algorithm that facilitates both open-ended and rapid discovery of the CTRNN creative search space.

## Author Keywords

Continuous Time Recurrent Neural Networks, Evolution, Novelty Search, Audio Synthesis

## 1. INTRODUCTION

Evolutionary algorithms (EAs) are a powerful optimisation technique based on Darwinian theory [9], grounded in the idea that the incredible novelty, diversity and complexity created by evolutionary processes in nature can be modelled in computers and applied to many different domains such as music. There are a variety of different compositional tools described within the literature that explore the creative possibilities of this type of machine learning, such as 'Eden' [16], 'MutaSynth' [2], 'Darwin Tunes' [15] and 'Synthbot' [26]. However, when we look at the various tools that are readily available to musicians, few EA-based examples exist. A possible cause for this situation is discussed by McCormack [17], who identifies that researchers applying evolutionary principles to art are "...trying to find the aesthetically satisfying needle from the data haystack of computation and algorithm" [17, p.424]. A contributing factor to this problem is the difficulty of finding an EA genotype (encoded representation of the phenotype or problem solution) structure that offers both variety in its search space and exploitability in realistic timeframes.

In prior work [11, 10], we have proposed the use of Continuous Time Recurrent Neural Networks (CTRNNs) as a possible candidate EA genotype for the generation of musical structures, which perform reasonably well in light of the 'needle in a haystack' problem. Much is known about CTRNNs and their capacity for extensive temporal dynamics [5, 1]. However, little research exists on exploring their potential as a low-level generator of musical structures such as audio waveforms. A possible cause for this is their idiosyncratic behaviour, which makes it hard to discover CTRNN configurations that could be useful to musicians without the aid of some form of machine learning. As CTRNNs have a compact structure capable of producing a vast range of temporal dynamics, they could provide an ideal genotype structure for an EA. By utilising this method, we could potentially discover and navigate the CTRNN musical search space in an effective way, a prospect that could facilitate a foundation for a co-creative tool that allows musicians to easily explore a large space of interesting audio possibilities without needing an expert level of experience required by many existing sound generation tools such as a modular synthesiser. Our prior work has supported this claim, as we have shown that EAs can be used to explore the musical search space of CTRNNs, which can be evolved to produce different timbres such as that of a Clarinet and a Sawtooth waveform. However, through these experiments, we are still yet to find an EA design that will be effective in achieving both thorough and open-ended discovery of the CTRNN search space.

Initially, an interactive EA was adopted allowing explicit control over the aesthetic trajectory of the creative search process without needing to define generalised targets for the algorithm to search for. However, the manual process of sifting through each individual of the population for assessment can be fatiguing, limiting population size and the number of generations that can be realistically explored [3]. A target-based EA approach was also explored and removed the need for constant human evaluation, allowing more rapid navigation of the CTRNN search space. However, the necessity for a specific predefined algorithm targets inhibited possibilities for the open-ended discovery achieved using an interactive approach.

This research [11, 10] has led us to explore alternatives to interactive and target-based EA designs, specifically Novelty Search (NS). Differing from the interactive and target-based algorithm structures used previously, predefined evolutionary targets are not required as the algorithm simply looks for novel candidates in relation to an existing population, encouraging open-ended discovery. This evolutionary process is also unsupervised, meaning there is no need for the cumbersome human evaluation of each individual.

*NIME'18,* June 3-6, 2018, Blacksburg, Virginia, USA.

We expect to find that this initial investigation into NS will provide solutions to some of the problems encountered in our prior research when adopting interactive and target-based EA approaches. We also foresee that this approach will raise future research question such as how best to algorithmically represent the aesthetic closeness of individuals in order to measure their novelty as well as how the resulting population can then be utilised by musicians. A code repository for this project can be found at `github.com/steffanianigro/plecto` and audio examples of various CTRNN configurations can be found at `nime2018.steffanianigro.com`

## 2. BACKGROUND

### 2.0.1 Evolutionary Algorithms

Evolutionary Algorithms are based on Darwinian theory, with evolutionary change a result of the fittest of each generation surviving and passing on the traits that made them fit [9]. This process starts with the random generation of an initial pool of candidate solutions. This population is then evolved through various iterations of an evolutionary process during which each individual's fitness is evaluated in relation to a selective pressure. The fittest individuals are bred through a crossover process during which genetic traits from both partners are passed onto their offspring. There is also a chance of mutation, achieving genetic drift or exploration outside the current gene pool. The new individuals replace a number of the least fit existing individuals and the process is repeated [23]. Asexual reproduction is also possible, with mutation being the only force for evolutionary change.

The selective pressure of an EA usually exists as an automated fitness function, evaluating individuals according to a criterion that is encoded into the system. However, in creative situations, it is difficult to define explicit fitness functions for audio phenotypes that can identify subjective musicality [22]. An alternative approach involves human evaluation of the population. The predominant paradigm for this approach sees the user take the role of a 'pigeon breeder', acting as a selective pressure in an artificial environment [4]. Furthermore, this approach shifts the EA into a new domain, as it "... is no longer a tool for finding optimal solutions, but instead becomes a vehicle for creative exploration" [25, p. 31], opening up possibilities for the unbound innovation seen in natural evolution [13]. These interactive algorithms work well in certain situations but tend to be less effective when used to explore large search spaces. This is due to the manual process of hand evaluating each individual within the population creating user fatigue, possibly inhibiting discovery of the most optimal creative solution.

Of most interest to the authors and what we expect will be the most effective in the exploration of musically interesting CTRNNs is novelty search (NS). NS is an evolutionary process during which an individual's fitness is not measured by closeness to a specific target but by how novel the individual is compared to the other individuals that have already been discovered [14]. A tool for open-ended exploration results that alleviates cumbersome human evaluation as well as the restrictiveness of target-based approaches in which only individuals with a high fitness can be further explored [13]. These characteristics could be advantageous for our use, being both efficient in exploring large search spaces and unbiased in the search for creative outputs. However, this approach comes with the challenge of computationally estimating novelty in a way that is also perceivable by a human listener.

### 2.0.2 Continuous Time Recurrent Neural Networks

Artificial Neutral Networks (ANNs) have been used for many different functions in music, from beat tracking algorithms [12] to artificial composers that can extract stylistic regularities from existing pieces and produce novel musical compositions based on these learned musical structures [18]. Bown and Lexer [5] offer another application, proposing the use of CTRNNs to create autonomous software agents that exhibit musicality. Bown and Lexer also outline that another suitable use for CTRNNs is in the audio signal domain, a prospect which inspired this research.

A notable example of similar work is discussed by Ohya [20], who describes a system that trains a Recurrent Neural Network to match an existing piece of audio. The network structure can then be manipulated to synthesise variants of the original sound. Eldridge [8] provides another example, exploring the use of Continuous Time Neural Models for audio synthesis. Within this research, we further explore the capabilities of using CTRNNs for the generation of musical structures.

CTRNNs are nonlinear continuous dynamical systems that can exhibit complex temporal behaviours [1]. They are an interconnected network of computer-modelled neurons, well suited to produce audio and rhythmic output as various configurations result in smooth oscillations that can be used to construct audio waveforms or trigger musical events. Each neuron of a CTRNN is typically of a type called the *leaky integrator*. The internal state of each neuron is determined by the differential equation (1),

$$\tau_i(dy_i/dt) = -y_i + \sum W_{ij}\sigma(g_j(y_j - b_j)) + I_i \qquad (1)$$

where $\tau_i$ is the time constant, $g_i$ is the gain and $b_i$ is the bias for neuron $i$. $I_i$ is any external input for neuron $i$ and $W_{ij}$ is the weight of the connection between neuron $i$ and neuron $j$. $\sigma$ is a *tanh* non-linear transfer function [5].

The behaviour of a neuron is defined by three parameters - gain, bias and time constant - and each neuron input has a weight parameter [5]. CTRNNs are continuous, recurrent, and, due to their complex dynamics, are typically trained using an EA. For this research, we adopt a fully connected CTRNN, meaning that the neurons in the hidden layer are all connected and the input layer has a full set of connections to the hidden layer. Each hidden neuron also has a self connection. The output or activation of each neuron is calculated using a *tanh* transfer function and produces outputs within the range of -1 and 1.

## 3. NS ALGORITHM DESIGN

Similar to our prior work [10], we adopt an EA structure based on the *opt-aiNet* algorithm by de Castro and Timmis [7]. It is a model that can maintain many individual candidate solutions to a problem, each evolved independently of one another. This means the algorithm can provide not only the global optimum, but many of the local optimum as well. Within this paper, we use multi-objective optimisation so that we can measure novelty through multiple isolated audio metrics that cannot be effectively combined into a single measure. The *opt-aiNet* algorithm is particularly useful for multi-objective optimisation because many Pareto optimal [24] candidates can be maintained.

Differing from more conventional EA structures, the *opt-aiNet* algorithm model incorporates sub-populations, each locally optimised. These sub-populations are generated by cloning and mutating each member of the EA's current population, with mutation rates inversely proportionate to the parent individual's fitness. The fittest individuals of each sub-population then replace the parent individual. This EA model also discourages convergence on a specific area of the search space using a population suppression mechanism. Once the population stagnates (when the difference between fitness errors over time fall below a predefined threshold), individuals of the main population are compared using a distance metric, after which the individuals with a close similarity are removed (and the fittest individuals are maintained). A number of randomly generated individuals are then introduced into the population (the population size can vary dynamically) to facilitate thorough exploration of the EA's search space [21]. In this paper, fitness refers to the novelty of individuals and is measured against a persistent population that exists outside of each algorithm iteration. This population will gradually grow as more and more individuals pass the novelty threshold. Below is an outline of each stage of the NS process.

1. Randomly initialise the main population for current iteration (not the persistent population).
2. While the stopping criterion is not met, continue, else save individuals to a database containing the persistent population for the next algorithm iteration. Go to step 1.
   (a) Calculate the fitness of each individual in the main population. The fitness of these individuals is measured against the persistent population.
   (b) Generate a number of clones for each individual, creating sub-populations.
   (c) Mutate each clone inversely proportionate to its parent's fitness (fitter individuals are mutated less).
   (d) Determine the fitness of individuals within each sub-population including the parent individual and remove all but the fittest, which replaces the parent individual in the main population.
3. Re-calculate the fitness of each individual in the main population after the fittest individuals of the sub-populations replace their parents. This is necessary to re-establish the fitness standing of the main population.
4. Calculate the average fitness of the individuals and if the population stagnates, continue to steps 5 else go to step 2.
5. Determine the highest affinity individuals (similar phenotype) and perform population suppression to avoid redundancy whilst maintaining the fittest individuals.
6. Introduce a number of randomly generated individuals.
7. Re-calculate the fitness of each individual in the main population after suppression and addition of random individuals. This is necessary to re-establish the fitness standing of the main population. Go to step 2.

The population for the *opt-aiNet* algorithm used in this paper is initiated with 10 random individuals. During each iteration, 5 clones are produced for each individual in order to create 10 independent sub-populations. The threshold dictating the chance of mutation for each parameter is calculated according to equation (2)

$$a = (1/\beta)\exp(-f*) \tag{2}$$

where $\beta$ is a parameter that controls the decay of the inverse exponential function and $f*$ is the fitness of the parent individual normalised within the interval of [0..1]. The parameter value to be mutated is calculated according to equation (3)

$$C' = c + aN(0,1) \tag{3}$$

where $c$ is a parameter value of a parent cell, $C'$ is the mutated parameter value, $a$ is calculated according to equation (2) and $N(0,1)$ is a Gaussian random variable with a mean of 0 and standard deviation of 1.

Each iteration of the *opt-aiNet* algorithm ends once the most novel individual of the main population passes a novelty threshold, meaning it is unique enough compared to the already exisiting individuals in the persistent population. This novel individual is then added to the persistent population, ready for the next iteration.

## 3.1 Comparison Metrics

Within this experiment, the EA will use a multi-objective function to measure novelty. In [10], we have explored the use of Mel-Frequency Cepstral Coefficients (MFCCs) and dominant frequency as metrics to calculate the spectral similarity of two individuals. This approach proved effective, therefore we adopt these metrics as well as spectral centroid in order to compare the novelty of individuals within the population. These metrics will be arranged in various combinations to determine which one best facilitates thorough discovery of the CTRNN search space. Effectively incorporating all aspects of a CTRNNs audio output in a phenotype representation exceeded the scope of this study. Therefore, we have not used any metrics to capture temporal variation such as rhythm, focusing on pitch and timbre so as not to overcomplicate the experiment. However, temporal metrics will need to be incorporated in a real world use case to capitalise on the rhythmic capabilities of CTRNNs.

These metrics are calculated from a frequency domain description of the audio being analysed, produced by applying a Fast Fourier Transform (FFT) to an 8192 sample window of audio after a Hamming windowing function is applied. The dominant frequency of the audio is calculated by identifying the frequency bin with the highest magnitude and the MFCCs are calculated as described in [27]: the FFT magnitudes are passed through a 42 component Mel filter bank spaced in the range of 20 to 22,050Hz, the 42 outputs of which are then transformed using a Discrete Cosine Transform. The first 12 of these 42 coefficients are kept and used for timbral comparison in this paper. spectral centroid is calculated by taking the mean frequency of the audio spectrum [6].

During an iteration of the EA, these audio metrics are extracted for each individual in the population. A novelty rank is assigned to each metric of each individual, calculated by finding the individual's nearest neighbour for each metric in the exisiting persistent population and comparing the result to other individuals of the iteration's population. The similarity of two dominant frequency or spectral centroid audio metrics is calculated by taking the absolute value of their difference. The similarity of two MFCCs is calculated using a Dynamic Time Warping (DTW) algorithm [19] with a Euclidean distance metric. The DTW algorithm is used for future proofing in case MFCCs are compared with different numbers of coefficients. Once these ranks are assigned for each metric, they are summed to measure the overall novelty of the individual. The EA is run until the most novel individual in the iteration's population passes a novelty threshold, after which it is added to the main persistent population. This process is repeated until the desired number of individuals are found or the novelty of new individuals can no longer exceed the novelty threshold (all novel individuals have been found).

# 4. EXPERIMENT

## 4.1 Description

This experiment is an initial investigation into the effectiveness of the described novelty search algorithm. We hope to answer two main questions when conducting this study. Firstly, we question whether this algorithm will outperform more mundane random search in finding a population of 100 CTRNN configurations that exhibit a variety of audio characteristics. Secondly, we hope to further understand what combination of audio metrics best facilitate the discovery of 100 novel audio CTRNNs, meaning a human will be able to easily differentiate between the phenotype representations of population individuals. In [10], we use both dominant frequency and MFCC metrics to compare the similarity of two sounds. MFCCs are pitch independent, therefore Dominant frequency was used to push the EA population towards the selected target sound's pitch. In this experiment, we are more concerned with discovering the CTRNN timbral space, as once interesting timbres are discovered, it is easy to shift their pitch during post-production of the audio because we are mainly dealing with monophonic sounds. Therefore, the most effective metrics may differ from our prior work.

In order to gain a basic understanding of the CTRNN space, we discovered 100 CTRNN configurations using plain random search. From prior studies [11, 10], we have found that the majority of randomly discovered CTRNNs do not exhibit musical behaviour, with outputs consisting of inaudible frequency content or lacking any persistent oscillations. This initial search will help confirm these prior observations and provide a base for comparing the populations discovered using NS.

For the first iteration of the NS algorithm, we only use the MFCC comparison metric to measure the novelty of individuals. This focus on timbre over specific pitch content will identify whether we can use a single timbral measure to explore the CTRNN audio search space, or if it will result in a lack of frequency diversity comparable to random search, with many individuals discovered being inaudible or musically undesirable. We also run two more iterations of the algorithm; first adding the dominant frequency metric, and then both the dominant frequency and spectral centroid metrics. These resulting populations will help identify the effects of including more frequency centric novelty metrics.

The populations created by random search and the three iterations of the NS algorithm are compared, allowing us to draw conclusions about the NS algorithm design and its effectiveness. We plot the MFCCs of all individuals in each population to compare timbral variation. The X axis represents the Mel Frequency Bins and the Y axis represents their cepstral coefficients. The closeness of the plotted lines correlate to the timbral closeness of the individuals. Therefore, lots of timbral variety in a population will be indicated by the variety of shapes and positions of the plotted lines in each graph. We will also plot the dominant frequency and spectral centroid values of each population on a scatter plot, providing an insight into the spread of discovered frequency content.

We expect to find that the novelty search algorithm with the single MFCC comparison metric produces individuals with timbral variety. However, similarly to the randomly discovered CTRNN configurations, individuals will default to non audible frequencies that will be of little use to musicians. We also expect that once dominant frequency and spectral centroid metrics are combined with the MFCC metric, the timbral space will remain similarly diverse. However, there will be a greater spread of frequency content within the population, producing CTRNN configurations that could be more valuable in musical contexts.

## 4.2 Results

Random generation was used to discover 100 CTRNN configurations and are plotted in Figures 1 and 2. Confirming our prior observations, the majority of CTRNN configurations either saturated (the CTRNN produces a consistent output of -1 or 1) or oscillated at low frequencies, behaviours which are not very useful for musicians unless they are used as LFOs or control signals. Furthermore, not much timbral variety is evident in the population, meaning many of the CTRNNs produced similar temporal behaviour and would not be creatively engaging for musicians.

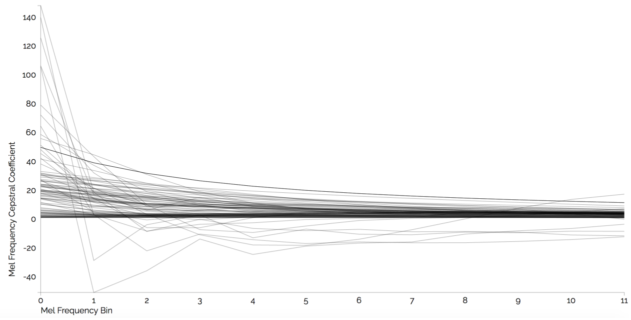

**Figure 1: MFCCs of 100 individuals discovered using random search.**

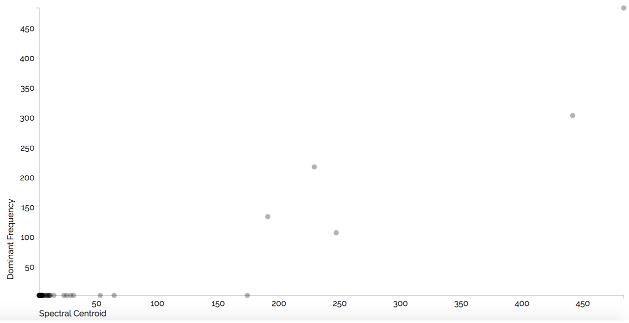

**Figure 2: Frequency spread of 100 individuals discovered using random search.**

These results contrast to the populations produced by the NS algorithm described in this paper, shown in Figures 3, 4, 5, 6, 7 and 8. Figures 2 and 3 depict the NS algorithm's output when only the MFCC metric was used to measure audio novelty. Figure 3 shows much more timbral variety in the population than achieved using plain random search. However, similarly to Figure 2, we do not see much spectral variety in Figure 4, with the majority frequency content of many individuals dwelling in inaudible frequency bins.

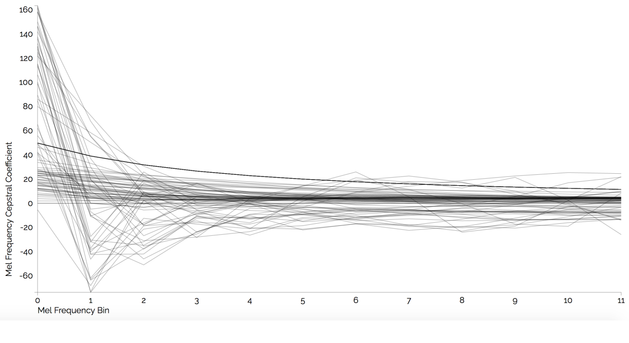

**Figure 3: MFCCs of 100 individuals discovered using NS with the MFCC comparison metric.**

Figures 5 and 6 depict the output of the NS algorithm that uses both the MFCC and dominant frequency metrics to measure audio novelty. In Figure 5, we can see a similar spread of timbral novelty. However, as we have shifted

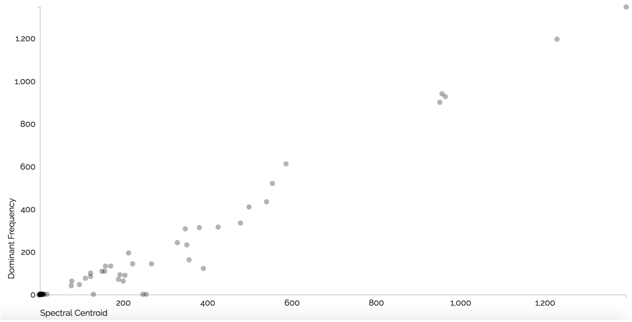

**Figure 4: Frequency spread of 100 individuals discovered using NS with the MFCC comparison metric.**

the novelty weighting from a purely timbral focus, we can see a larger spread of frequency content in the discovered population, represented in Figure 6. These results confirm that combining timbral and frequency based metrics aid in achieving a population with a greater diversity of audio features.

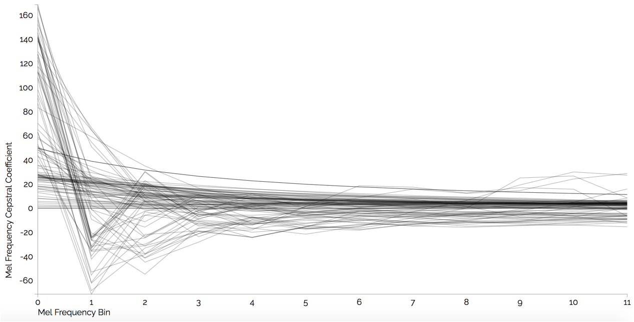

**Figure 5: MFCCs of 100 individuals discovered using NS with MFCC and dominant frequency comparison metrics.**

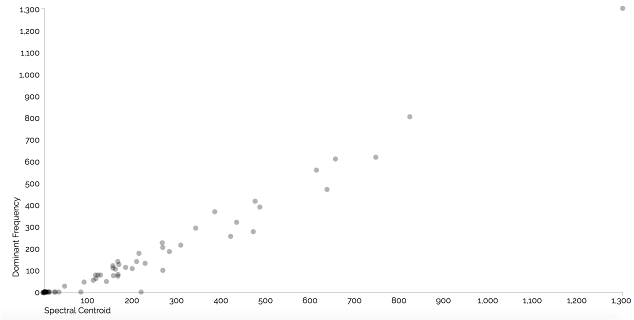

**Figure 6: Frequency spread of 100 individuals discovered using NS with MFCC and dominant frequency comparison metrics.**

A further confirmation of this observation is evident when the MFCC metric is combined with both dominant frequency and spectral centroid metrics. Figure 7 depicts the various MFCCs of each individual in the population, highlighting similar timbral variety than evident in Figures 3 and 5. We can also see less clustering at the lower bottom left hand corner of Figure 8, compared to Figures 2, 4 and 6. This means that the algorithm has achieved similar timbral variety yet has produced a greater variety of frequency content in the population, achieved by adding another metric defined by the clustering of frequencies in the audio being analysed, further shifting the algorithm's novelty weighting away from a purely timbral focus.

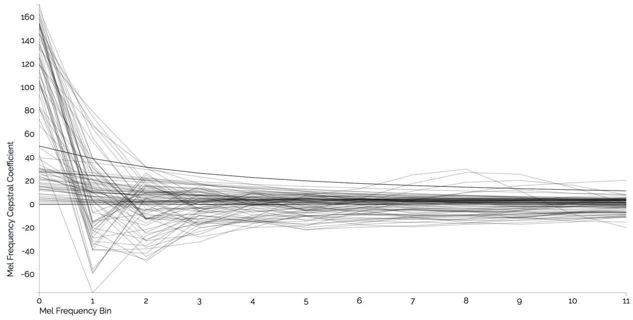

**Figure 7: MFCCs of 100 individuals discovered using NS with MFCC, dominant frequency and spectral centroid comparison metrics.**

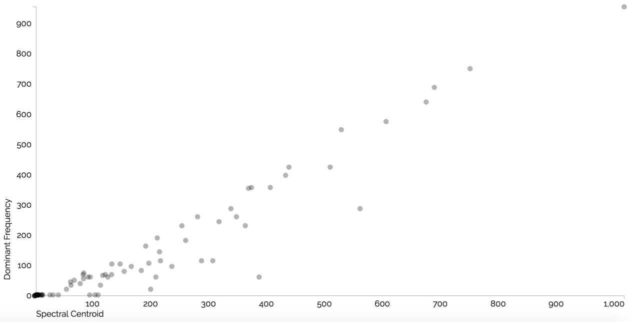

**Figure 8: Frequency spread of 100 individuals discovered using NS with MFCC, dominant frequency and spectral centroid comparison metrics.**

Overall, the most effective approach was the NS algorithm with a combination of MFCC, dominant frequency and spectral centroid metrics, as individuals were discovered with both timbral variety and a large range of frequency content within the audible domain. A downside to the incorporation of these additional metrics was that the algorithm's run time increased by a substantial amount. As observed in Figures 1 and 2, the CTRNN configurations seem to gravitate to lower frequency domains which could be a cause for this decrease in efficiency, as it takes longer to discover CTRNN configurations that oscillate at higher frequencies. In future experiments we hope to further increase the algorithm's efficiency and look towards different hardware integrations (GPUs) to allow much larger population and sub-population sizes, more computationally intense metrics (running FFTs over larger sample sizes for better frequency resolution) and higher novelty thresholds.

## 5. CONCLUSION

Contrasting to our prior work, NS offers a means to navigate the audio possibilities of CTRNNs in an effective, open-ended manner. The NS algorithm discussed in this paper out-performed random search and created a range of CTRNNs that spanned a larger range of frequency and timbral realms. Listening to samples of the various CTRNN populations confirmed the results of the above experiments. CTRNNs discovered through NS feature a larger range of timbres and frequency ranges, samples of which can be heard at `nime2018.steffanianigro.com`. In contrast, most randomly discovered individuals exhibited no desirable audio qualities. However, even though there is variety in the CTRNNs discovered using NS, some individuals do sound similar. This highlights the need to refine the algorithm metrics to better capture nuances of how humans perceive audio novelty in the phenotype space. This may require the introduction of different metric weightings, favouring measures more relevant to the way humans perceive audio. The lack of temporal metrics could also be a contributor to this issue as temporal variation is an important part of timbral identification. Introduction of temporal metrics will also help capitalise on the rhythmic potential of CTRNNs. Furthermore, the current NS algorithm only deals with a fixed input for all CTRNN configurations. If these configurations are to be used in real world contexts, consistent behaviour over multiple input types is required. For example, a CTRNN configuration will not be very useful if it only behaves in interesting ways under very specific circumstances. They should be versatile enough to be useful in multiple musical contexts, an issue we will address by measuring an individual's novelty over a range of CTRNN input types instead of just the one used in the described experiments. We will also further expand on the interaction design aspect of our prior work [11] and develop an effective interface design that provides access to the resulting novel audio CTRNNs of these studies. A qualitative evaluation stage will also follow during which we will conduct user studies to gain insight into the effectiveness of our design and how it can be further improved in future iterations of our system. The code repository for this project can be found at `github.com/steffanianigro/plecto`.

## 6. REFERENCES

[1] R. D. Beer. On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509, 1995.

[2] R. Berry and P. Dahlstedt. Artificial life: Why should musicians bother? *Contemporary Music Review*, 22(3):57–67, 2003.

[3] J. Biles, P. Anderson, and L. Loggi. Neural network fitness functions for a musical iga. 1996.

[4] O. Bown. Ecosystem models for real-time generative music: A methodology and framework. In *International Computer Music Conference (Gary Scavone 16 August 2009 to 21 August 2009)*, pages 537–540. The International Computer Music Association, 2009.

[5] O. Bown and S. Lexer. Continuous-time recurrent neural networks for generative and interactive musical performance. In *Applications of Evolutionary Computing*, pages 652–663. Springer, 2006.

[6] G. Carpentier, D. Tardieu, J. Harvey, G. Assayag, and E. Saint-James. Predicting timbre features of instrument sound combinations: Application to automatic orchestration. *Journal of New Music Research*, 39(1):47–61, 2010.

[7] L. N. de Castro and J. Timmis. An artificial immune network for multimodal function optimization. In *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on*, volume 1, pages 699–704. IEEE, 2002.

[8] A. Eldridge. Neural oscillator synthesis: Generating adaptive signals with a continuous-time neural model. 2005.

[9] P. Husbands, P. Copley, A. Eldridge, and J. Mandelis. An introduction to evolutionary computing for musicians. In *Evolutionary computer music*, pages 1–27. Springer, 2007.

[10] S. Ianigro and O. Bown. Investigating the musical affordances of continuous time recurrent neural networks. In *Proceedings of the Seventh International Conference on Computational Creativity*, 2016.

[11] S. Ianigro and O. Bown. Plecto: A low-level interactive genetic algorithm for the evolution of audio. In *Evolutionary and Biologically Inspired Music, Sound, Art and Design*, pages 63–78. Springer, 2016.

[12] A. J. Lambert, T. Weyde, and N. Armstrong. Perceiving and predicting expressive rhythm with recurrent neural networks. 2015.

[13] J. Lehman and K. O. Stanley. Exploiting open-endedness to solve problems through the search for novelty. In *ALIFE*, pages 329–336, 2008.

[14] J. Lehman and K. O. Stanley. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation*, 19(2):189–223, 2011.

[15] R. M. MacCallum, M. Mauch, A. Burt, and A. M. Leroi. Evolution of music by public choice. *Proceedings of the National Academy of Sciences*, 109(30):12081–12086, 2012.

[16] J. McCormack. Eden: An evolutionary sonic ecosystem. In *Advances in Artificial Life*, pages 133–142. Springer, 2001.

[17] J. McCormack. Facing the future: Evolutionary possibilities for human-machine creativity. In *The Art of Artificial Evolution*, pages 417–451. Springer, 2008.

[18] M. C. Mozer. Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multi-scale processing. *Connection Science*, 6(2-3):247–280, 1994.

[19] L. Muda, M. Begam, and I. Elamvazuthi. Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques. *arXiv preprint arXiv:1003.4083*, 2010.

[20] K. Ohya. A sound synthesis by recurrent neural network. In *Proceedings of the 1995 International Computer Music Conference*, pages 420–423, 1995.

[21] J. Timmis and C. Edmonds. A comment on opt-ainet: An immune network algorithm for optimisation. In *Genetic and Evolutionary Computation–GECCO 2004*, pages 308–317. Springer, 2004.

[22] N. Tokui and H. Iba. Music composition with interactive evolutionary computation. In *Proceedings of the 3rd international conference on generative art*, volume 17, pages 215–226, 2000.

[23] D. Tzimeas and E. Mangina. Dynamic techniques for genetic algorithm–based music systems. *Computer Music Journal*, 33(3):45–60, 2009.

[24] D. A. Van Veldhuizen and G. B. Lamont. Evolutionary computation and convergence to a pareto front. In *Late breaking papers at the genetic programming 1998 conference*, pages 221–228. Citeseer, 1998.

[25] S. Woolf and M. Yee-King. Virtual and physical interfaces for collaborative evolution of sound. *Contemporary Music Review*, 22(3):31–41, 2003.

[26] M. Yee-King and M. Roth. Synthbot: An unsupervised software synthesizer programmer. In *Proceedings of the International Computer Music Conference, Ireland*, 2008.

[27] M. J. Yee-King. *Automatic sound synthesizer programming: techniques and applications*. University of Sussex, 2011.