Soft Comput (2012) 16:2057–2070
DOI 10.1007/s00500-012-0873-x

FOCUS

Automated evolutionary synthesis matching

Advanced evolutionary algorithms for difficult sound matching problems

Thomas Mitchell

Published online: 19 June 2012

Abstract This paper discusses the subject of automatic evolutionary sound matching: systems in which evolutionary algorithms are used to automatically derive the parameters of a synthesiser to produce a sound that matches a specified target sound. The paper describes prior work and identifies the principal causes of match inaccuracy, which are often due to optimiser limitations as a result of search space problem difficulty. The components of evolutionary matching systems contributing to problem difficulty are discussed and suggestions as to how improvements can be made through problem simplification or optimiser sophistication are considered. Subsequently, a novel clustering evolution strategy is presented which enables the concurrent optimisation of multiple distinct search space solutions, intended for the purposes of sound matching with standard frequency modulation (FM) synthesisers. The algorithm is shown to outperform standard multi-membered and multi-start $(1 + 1)$ evolution strategies in application to different FM synthesis models for static and dynamic sounds. The comparative study makes use of a contrived matching method, which ensures that results are not affected by the limitations of the matching synthesiser.

Keywords Evolutionary computation · Evolutionary sound matching · Frequency modulation synthesis · Clustering evolutionary algorithms · Evolution strategy

T. Mitchell (✉)
Computer Science and Creative Technologies,
University of the West of England, Bristol BS16 1QY, UK
e-mail: tom.mitchell@uwe.ac.uk

1 Introduction

Modern technology has had a profound effect on the structure, form and performance of music. Powerful and inexpensive general-purpose computers have made electronic musical apparatus widely available to amateur and professional composers alike. The audio synthesiser has played, and continues to play an important role in the development of modern music, enabling composers to electronically recreate the sound of acoustic instruments, or to explore beyond the realms of the familiar, to create sounds previously unheard. There are a wide variety of synthesis techniques which can be used to create musical sounds across a considerable range of timbres. Effective control and navigation of a synthesiser's sound space requires expert knowledge of the underlying synthesis technique, which may draw from theoretical and/or experiential knowledge. The parameters which are used to shape the sound character are specific to the particular synthesis architecture being employed, and rarely relate to sound in human terms. Consequently, there is often a complex mapping between the dimensions of a synthesis parameter (or control) space, and the perceived sound character (or timbre) space.

If it were possible to relate the parameters of a synthesiser more directly to the user's intuitive understating of timbre, synthesiser control could become more transparently about sound creation rather than computer programming. The first step to achieving this is the development of a process which is able to map known sound qualities onto sound synthesis parameters. This requires a technique that can efficiently search a synthesis parameter space to identify configurations which achieve specific timbral characteristics. In recent times, researchers have experimented with evolutionary computation (EC) to facilitate

the autonomous matching of target sounds with a variety of different synthesis types. This valuable research has the potential to enable synthesisers to automatically self-program and to facilitate the development of new synthesiser interfaces that enable control in human terms.

In Sects. 2 and 3 of this paper an overview of automated evolutionary matching synthesis is provided. The components of the system which can lead to match inaccuracy are discussed and suggestions as to how match accuracy can be improved are considered. In Sect. 4, the properties of standard evolutionary algorithms (EAs) which can lead to suboptimal convergence are discussed. Throughout Sect. 5 a novel clustering evolution strategy (CES) is developed incorporating the notion of species within the standard evolution strategy (ES) model with the potential to increase sound match accuracy. In Sects. 6, 7, 8 and 9 an evolutionary frequency modulation (FM) synthesis matching system is developed and used to compare the performance of the developed optimiser with more standard algorithms. This paper extends the work presented previously in Mitchell and Creasey (2007).

2 Previous evolutionary sound matching work

When an evolutionary sound match is performed, the system is initially supplied with a target sound which is to be matched by the system. This target sound is analysed to extract a representation which enables the difference between potential synthesised matches and the original target to be quantified for the purposes of fitness assessment. The EA population is then initialised and evolved, in a cycle of variation and selection, to breed increasingly closer matches to the target. In general, population individuals comprise a complete set of synthesis parameters, which are mapped to a corresponding sound via the synthesiser for subsequent comparison with the target sound.

The earliest evolutionary sound matching systems were developed by Horner et al. (1993a, b) at the University of Illinois. In this work, Horner employed a genetic algorithm (Holland 1975) to evolve parameters of FM/wavetable synthesis to reproduce the sounds of acoustic musical instruments. Later evolutionary sound matching efforts included the work of Riionheimo and Välimäki (2003) applying the genetic algorithm to match target sounds using a plucked string physical model. Evolutionary algorithms have also been employed to grow modular synthesis circuits. For example, Garcia (2002) and Wehn (1998) evolved the arrangement and interconnection of synthesis graphs using genetic programming and genetic algorithms, respectively. More recently, matching experiments with genetic algorithms have been presented by Bozkurt and Yüksel (2011) in application to multiple-modulator FM synthesis; McDermott et al. (2008) in application to a modular subtractive synthesiser; and Yee-King and Roth (2008) in application to any available VSTi software synthesiser.

3 Match inaccuracy

Typically, the results of evolutionary synthesis matching experimentation is provided in the form of error values with respect to the chosen target sounds (McDermott et al. 2008), sometimes with the support of time and frequency domain plots to enable the differences between sounds to be compared visually (Yee-King and Roth 2011). Occasionally, subjective tests with human listeners are also included to verify the findings (Horner 1998; Mitchell 2010). When the matching system is found unable to evolve an accurate simulation of a target sound, it is often not clear as to whether this is due to the limitations of the matching synthesiser, or misbehaviour of the optimisation algorithm. As these components of the matching problem are rarely considered in isolation, it becomes difficult to examine the pathology of inaccurate matches.

Synthesiser limitations There is an abundance of potential synthesis techniques appropriate for sound matching by EC (Roads 1996). Each synthesis type offers distinctive characteristics and thus a constrained sound space. When an evolved sound match is inaccurate, a possible explanation might be that the matching synthesiser is simply incapable of reproducing the target, in which case it is hoped that the evolved sound represents the most accurate match available.

Optimiser limitations Another factor limiting match accuracy might be the capabilities of the underlying optimisation algorithm, where the EA is incapable of locating the optimal solution because the characteristics of the synthesis matching problem space lead to suboptimal convergence. This may be indicative of a 'difficult' problem space which features search space characteristics which are problematic for standard optimisation algorithms.

As the motivation guiding this field of study is sympathetic to synthesiser limitations, developing an understanding of the components limiting optimiser performance is important to the future developments of evolutionary sound matching. To improve the quality of sound matches, efforts must be made to simplify the problem and/or improve the performance of the optimisation algorithms.

3.1 Problem difficulty

When the fitness landscape of an optimisation problem comprises multiple distinct local optima surrounded by

regions of low-fitness noise, it is challenging for traditional EAs to sufficiently characterise the space before the population converges to a single optimum (Mahfoud 1995). Analysis of the synthesis matching problem space is difficult to perform as the characteristics of the sound matching problem alter significantly from one target sound to the next (Lim and Tan 1999). Consequently, it is difficult to draw conclusions that apply to every possible test case.

Preliminary search space analyses of wavetable and FM synthesis has been performed by Horner (1997). In that study, a set of randomly generated tones produced by each synthesis method were compared with a small selection of target sounds. The results indicated the availability of 'good' matches within each synthesis space and it was found that the matches were least abundant in the FM search space. Also included in Horner's analysis of FM was a one-dimensional visualisation, created by plotting the spectral difference measure against a single synthesis parameter. Comparable analysis of the alternative synthesis methods produced plots which contained significantly fewer local optima. From this study Horner concluded that a simple hill-climbing search strategy would be insufficient for successful exploitation of the multimodal FM parameter space.

McDermott et al. (2008), later performed an analysis by measuring *Fitness distance correlation* and *monotonicity* to compare the effects of different sound similarity measures on the problem difficulty to identify those measures that would be the least challenging to optimise. Furthermore, Yee-King and Roth (2011) used fitness landscape plots for visualising error as synthesis parameters are changed with respect to a reference tone. The surface clearly indicated the wide range of sounds available when using a simple FM synthesiser. Comparable landscape plots have been presented in previous work by Lim and Tan (1999) and Mitchell (2010).

3.2 Improving performance

If matching is to be performed from within the sound space limitations of a given synthesiser, effective navigation of its parameter space is necessary. Ideally, the optimisation algorithm should be capable of locating the optimal match for the associated synthesiser. In many complex synthesis spaces this is not possible without exhaustive search methods; however, efforts can be made to enable the evolution of more accurate matches by enhancing the performance of the optimisation algorithm such that it is more adept at producing accurate matches within multimodal search spaces. This is the main focus of this work: developing and applying more sophisticated EAs to produce accurate simulations of target sounds. This area of study has largely been overlooked so far in the literature. The following sections will discuss the propensity for standard EAs to converge to a single optimum in multimodal search spaces and then present the development of a novel niching ES, designed to concurrently optimise multiple distinct solutions within a single population.

4 Suboptimal convergence

Evolutionary algorithms have been shown to be robust, reliable and straightforward to implement even when there is very little a priori knowledge of the application domain. Consequently, there has been growing interest in the application of EAs to an ever-increasing range of parameter optimisation problems. However, despite its strengths, evolutionary optimisation is not without weakness: when used to optimise multimodal search spaces, traditional EAs are unable to maintain solutions to more than one optima, regardless of the population size (Mahfoud 1995). The primary reason that standard EAs often fail within these environments is endemic to their architecture. The model combines stochastic search operators, to explore the problem space, with selective operators, to exploit profitable regions. Consequently, the evolving population tends to rapidly focus on a single peak, which may be disadvantageous when the application domain is comprised of multiple high-fitness peaks, as is often the case within synthesis matching applications (Horner 1997; McDermott et al. 2008).

Central to the development of advanced EAs designed to operate within multimodal environments are the concepts of *niche* and *species*. Although loosely defined, the term species is used to refer to solutions that share similar characteristics, and niche to refer to the region within the search space that a species occupies. To ensure appropriate modification to the standard EA model, it is important to note the properties of traditional EAs that preclude the formation of species:

– *Recombination disruption* Recombination has the power to destroy, as well as unite, the beneficial traits of individuals. When used to optimise a multimodal problem space, traditional recombination, acting globally on the population, will attempt to blend genetic material from individuals representing independent search space peaks without bias. The corresponding recombinant will thus characterise some midpoint between contributing individuals, and is not guaranteed to occupy any of the peaks represented in the parental set. These disruptive effects of global recombination may be reduced by modifying the EA to ensure that mating only takes place locally between members of the same niche.

– *Optimistic selection* Traditional EA selection operators consider only fitness when identifying those members of the population to partake in recombination. As such, it is possible for a single adaptation with high relative fitness to dominate the population before other regions of the search space have been sufficiently explored. To enable the formation of species, it is required that the selection/replacement operators consider not only fitness, but also the location of each individual with respect to the rest of the population.

In the following section, a novel EA is proposed in which *k*-means clustering is included within the generational model of the ES to address the issues of recombination disruption and optimistic selection, thereby enabling the formation of population species and thus the concurrent optimisation of multiple search space optima.

5 Clustering evolution strategy (CES)

To partition the population into species, a clustering procedure can be employed to identify those individuals occupying the same locality within the search space. The pseudocode for this algorithm is shown in Fig. 1. The randomly initialised parent population is first partitioned into species using a clustering algorithm. Each cluster of individuals within the parent population is subsequently recombined, mutated and selected as parents for the next generation. New parents are then reclustered, recombined, mutated and so forth until the end condition is satisfied. The re-clustering of the parent population at the beginning of each generation is central to the success of this algorithm within multimodal environments, as it ensures that clusters converging on the same niche merge to form a single cluster. Remaining clusters will then be assigned to individuals placed elsewhere in the search space, promoting increased exploration and preventing the entire population from gravitating towards a single peak.

```
t = 0;
initialise(μ(t));
while begin
      cluster(μ(t), k);
      for i = 1 to k
            λ_i(t) = recombine(μ_i(t));
            mutate(λ_i(t));
            evaluate(λ_i(t));
            μ(t+1) += select(λ_i(t));
      for end
      t = t + 1;
while end
```

Fig. 1 Clustering evolution strategy pseudocode

5.1 *K*-means clustering

K-means analysis (MacQuee 1967) is a well-known unsupervised algorithm which is designed to identify structure within data samples. Because of its suitability for a variety of pattern recognition problems, *k*-means cluster analysis has found extensive use within fields of image processing, data compression, data mining, statistics and natural sciences. More recently, clustering is being incorporated into EAs to assist with the optimisation of multimodal search spaces. In the context of EC, the population constitutes the dataset, and cluster analysis provides a procedure for classifying population members into species. To ensure that population members are accurately grouped into diverse species, the *k*-means cluster centroids within the CES are initialised according to the furthest point algorithm (Gonzalez 1985).

5.2 Cluster-based recombination

To address the issues relating to recombination disruption identified above, two cluster-orientated recombination operators are proposed that prohibit mating between parents that do not belong to the same cluster:

– *Discrete recombination* Discrete recombination engenders offspring by copying alleles directly from randomly selected parents drawn from within the same cluster. Parents within a cluster are selected uniformly at random and parents belonging to other clusters cannot be selected.
– *Centroid recombination* Beyer (2001) has demonstrated theoretically that progress rates can be significantly improved by setting the number of parents that partake in recombination as high as possible. Intermediate recombination is then the process of assigning each offspring individual to the centroid of the parent population. Within the CES, this procedure is already performed for each cluster by *k*-means analysis. Therefore, centroid recombination automatically assigns the offspring from each cluster directly to the position of its parents' cluster centroid, removing the need for recombination of object parameters entirely. The process of cluster analysis is therefore intimately linked with the recombination operator.

5.3 Mutation

Mutation within the CES is provided by the derandomised self-adaptation method developed by Ostermeier et al. (1994) as it is tuned to operate effectively on individuals

rather than global populations, which is of benefit when the population is subdivided, as with the CES.

5.4 Restricted cluster selection

Restricted cluster selection addresses issues associated with optimistic selection identified above by drawing the fittest $\frac{\mu}{k}$ individuals from among offspring produced by each cluster, to form the parent population of the subsequent generation.

Throughout the remaining sections of this paper the FM synthesis method will be introduced, prior work reviewed and the performance of the CES will be assessed in comparison with two other ES-based optimisers in application to the FM synthesis matching problem.

6 Frequency modulation synthesis

Since the focus of this work is the derivation of parameters for standard synthesis techniques, it is necessary to choose one with which to work. Frequency modulation synthesis (Chowning 1973) has been chosen for the following reasons:

- FM synthesis presents a method for generating sound which has seen wide application in commercial systems, and thus represents a real-world synthesis technique. Since its introduction, there have been many attempts to simulate specific sound types with FM synthesis; see, for example, Delprat (1997), Risberg (1980) and Schottstaedt (1977). This provides a historical context for the sound matching problem.
- The FM synthesis space is non-linear. A synthesis model is considered to be non-linear when the perceived timbre does not change in a consistent and proportional manner as the synthesis parameters are varied; there is a complex parameter space mapping. For example, the linear incrementation of a single parameter may cause a sound to move through many dimensions of the timbre space with a complex trajectory. Moreover, this trajectory may be entirely different when other synthesis parameters are changed. For fuller description of these issues see Ashley (1986).
- The FM synthesis model is compact and efficient. With only a limited number of parameters, it is possible to generate a wide range of complex time-varying sound textures with as little as two sinusoid calculations, two multiplies and one addition for each synthesis sample (Roads 1996).

In what is termed simple FM, the instantaneous frequency of one sinusoidal oscillator is modulated by another. A diagram of the simple FM model is provided in Fig. 2.

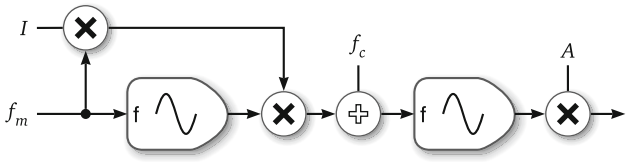

Fig. 2 Simple FM synthesis model

The instantaneous amplitude function for simple FM is given by:

$$e = A\sin(2\pi f_c t + I\sin(2\pi f_m t)) \quad (1)$$

where e is the modulated carrier output, A is the peak amplitude of the carrier, f_c and f_m are the carrier and modulator frequencies, respectively, and I is the modulation index. When I is assigned a value of zero there is no modulation, and the generated signal is a sine wave at frequency f_c. However, when $I > 0$, frequency partials are generated at frequencies $f_c \pm nf_m$, where n is an integer. The amplitudes of these partials are governed by the Bessel functions of the first kind and order n.

The desire to develop a systematic means by which FM synthesis can be employed to simulate real acoustic instruments has motivated a series of studies. Chowning's original paper initiates interest in this direction, providing example parameters that simulate brass, woodwind and percussive tones with the simple FM architecture. Schottstaedt (1977) later provided example parameters for simulating stringed instruments, including piano and violin tones. Subsequently, many researchers set out to develop a system to automatically derive FM synthesis parameters to reproduce particular target sounds. Early analytical efforts to automate sound design with FM were proposed by Delprat (1997), Justice (1979), Payne (1987) and Risberg (1980), although none of these methods are complete and often development is left as future work.

More recent advances in automated sound matching with FM synthesis have used EAs to optimise synthesis parameters. Horner's (1993) FM-matching algorithm optimises a set of static basis-spectra, produced by FM synthesis, which are dynamically recombined to simulate time-varying harmonic sounds. The amplitude envelopes for the basis-spectra are then determined by direct least-squares solution. The synthesis process is comparable with wavetable synthesis, with FM used in the production of the basis-spectra. The wavetable basis-spectra are generated by a special configuration of the simple FM model, known as formant FM, in which the modulator frequency is set to the fundamental of the target sound and the carrier frequencies are restricted to integer multiples thereof. Restriction of the carrier frequencies in this way ensures that only harmonic basis-spectra are considered. In a later study, the author of this paper compared the performance of several ES-based

EAs for matching a set of non-changing static target tones with the simple FM model (Mitchell and Pipe 2006). The EA designed for multimodal optimisation was found to produce the most accurate matches.

Interestingly, a restricted form of the FM synthesis matching problem has recently been adopted as a benchmark problem for testing EAs on real world problems (Das and Suganthan 2011); a testimony to the challenging nature of this problem.

7 Fitness measure

Within the evolutionary sound matching system proposed here, the fitness of each individual is determined by the following procedure:

1. Insert candidate solution into the FM model to synthesise a corresponding waveform.
2. Transform waveform into frequency domain representation by short-time Fourier transform (STFT).
3. Compute fitness by comparing the frequency domain representations of the target and synthesised candidate sounds using the relative spectral error metric.

The STFT is performed by dividing the target signal $x(n)$ into frames, which are then transformed into frequency domain data using the discrete Fourier transform (DFT):

$$X(m,k) = \sum_{n=0}^{N-1} w(n)x(n+ms)e^{-j\frac{2\pi kn}{N}} \tag{2}$$

where $X(m, k)$ is the STFT of the signal $x(n)$, with integers $m = 0, 1, 2, 3, \ldots$ and $k = 0, 1, \ldots, N-1$ referring to the frame index and frequency bin respectively. N is the DFT frame size, s is the step-size between successive time frames, and $w(n)$ is a window function.

To prevent unwanted artefacts due to discontinuities at frame boundaries, the Hamming window is employed (Miranda 2002), defined as:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \tag{3}$$

To capture the development of the frequency spectrum over time, multiple spectra are taken throughout the duration of the target sound. Previous matching efforts have utilised the complete set of short-time spectra, measuring the average error computed for all frames (Riionheimo and Välimäki 2003). However, since many musical sounds develop slowly with time, often only a small number of frames are required to sufficiently characterise the target sound (Beauchamp and Horner 2003). For the subsequent experimentation with time varying sounds, 10 frames of size 1024 are taken at uniform intervals throughout the duration of the target sound. For static tones, a single frame of size 1024 is taken.

To enable the difference between candidate and target sounds to be quantified, sound similarity is measured by computing the relative error between the spectra of the target and candidate sounds. This error measure, and variations thereof, has proved effective in previous evolutionary matching studies and offers an excellent balance between detail and execution speed; see, for example, the work of Garcia (2002), Horner et al. (1993), Riionheimo and Välimäki (2003) and Wehn (1998).

The relative spectral error is computed by accumulating the normalised difference between each frequency component of the candidate spectrum against their corresponding components in the target spectrum. The relative spectral error is defined as:

$$E = \frac{1}{N_{\text{frames}}} \sum_{t=1}^{N_{\text{frames}}} \sqrt{\frac{\sum_{b=0}^{N_{\text{bins}}} (T_{tb} - S_{tb})^2}{\sum_{b=0}^{N_{\text{bins}}} T_{tb}^2}} \tag{4}$$

where E is the relative error, T is a vector of target spectrum amplitude coefficients, S is a vector of synthesised candidate spectrum amplitude coefficients, N_{frames} is the number of static spectra analysed over the duration of the sound and N_{bins} is the number of frequency bins produced by spectrum analysis. A relative error of zero indicates an exact match, and comparison between the target sound and silence results in an error of 1.0. Studies performed by Beauchamp and Horner (2003, 2006) with acoustic musical instrument sounds have established that the relative spectral error correlates well with the average discrimination data extracted from human listeners. Furthermore, when the relative error was calculated using less than 10 frames of each sound, the correlation compared favourably with those attained when the entire frame set was used.

8 Choice of target sounds

When designing experiments to test the performance of evolutionary sound matching systems, a principal aim is to measure the ability of the EA to access all regions of the synthesis space and consistently identify accurate matches. In previous work, optimisation performance was frequently quantified by measuring the quality of the optimised solutions when matching arbitrary target sounds. Target sounds may be real dynamic sounds originating from acoustic instruments (Horner et al. 1993), or simple periodic tones generated by additive synthesis (McDermott et al. 2005). An alternative method is available, whereby performance is measured by the ability of an EA to match contrived targets, generated by the matching synthesiser (Mitchell and Creasey 2007). This approach is inspired by

the early FM matching work presented by Justice (1979) and Payne (1987), and has been adopted in previous studies by McDermott et al. (2008), Riionheimo and Välimäki (2003) and Yee-King and Roth (2011).

A contrived target is a sound or tone that originates from within the search space of, and is generated by, the matching synthesiser. Contrived target sounds provide two significant advantages over experimentation with non-contrived alternatives, both related to easing the task of measuring the performance of the matching system:

- It is simple to determine when an optimal solution has been evolved as the match and target will be identical, achieving a relative spectral error of zero. If non-contrived target sounds are chosen as test specimens, confirmation of optimal convergence is not as easy. For example, the matching synthesiser may not be capable of exactly reproducing a particular target sound recorded from a real acoustic instrument, in which case a match delivering a relative error of zero cannot be achieved. In these circumstances an optimal match may only be confirmed when an exhaustive search yields no better result, an approach that becomes infeasible as the problem dimensionality increases.
- Producing targets by randomly generating points within the synthesis space ensures that the test set constitutes a diversity of search space positions, and thus assesses performance on a variety of search space landscapes, as the topology of the landscape is dependent upon the properties of the target sound. Moreover, repeated matching of random contrived targets indicates whether it is possible to access all regions of the search space.

The results from experimentation with contrived targets may then be used as an indicator of a matching system's ability to evolve the most accurate match of any arbitrary target sound.

In the subsequent experimentation the contrived matching method is used to assess and contrast the performance of different ES-based optimisers including the CES presented above. Subsequent to this comparison, matches with acoustic instrument sounds are used to confirm the performance of the matching method in application to real-world sounds.

9 A comparison of optimisation algorithms

Three EAs are tested and compared in the following experimentation:

1. Evolution strategy (ES)
2. Multi-start (1 + 1) evolution strategy (MSES)
3. Clustering evolution strategy (CES).

Algorithm 1 is the standard multi-membered evolution strategy developed originally by Rechenberg (1965) and Schwefel (1995). Algorithm 2 is a variant on the basic two-membered (1 + 1) ES as defined also by Rechenberg (1965). Multiple instances of this algorithm are evolved concurrently (occasionally this algorithm is referred to as a multi-start hill-climber (Streichert et al. 2000). Each (1 + 1) ES mutates its object parameters isotropically according to a single mutation step-size, which is adapted by the 1/5th rule (Schwefel 1995). Algorithm 3 is the CES proposed in Sect. 5.

Experimentation is divided into two parts. The first part considers the matching of non-changing static tones, with an FM synthesis model in which the parameters remain stationary throughout the synthesis process. The second part is concerned with matching time-varying, dynamic sounds, by allowing certain parameters to change as synthesis takes place. This terminology is maintained henceforth, referring to timbres with a constant spectral form as static tones, and timbres in which the spectrum changes over time as dynamic sounds. The division of the experimental results into these two parts represents natural progression in tackling the synthesis matching problem, and corresponds directly to the chronological development of this work.

To ensure parity across all experiments, consistent algorithmic parameters and operators are fixed for all test cases. Indicated results are calculated by the mean average of 30 runs, matching 30 randomly generated contrived targets. All statements indicating an inequality of means are confirmed to be significant at the 0.05 level using a one-way ANOVA with a post-hock Games–Howell pairwise tests, unless stated otherwise. Each algorithm is tested when matching the same target set and populations are initialised with the same random data points, enabling observed differences between results to be attributed to the search properties of the EAs. Each algorithm runs for exactly 50 generations, except for the MSES test cases, which run for exactly the same number of fitness evaluations (70,000). Wherever applicable, both intermediate (or centroid for the CES) and discrete recombination are employed. For the purposes of brevity only results from experimentation with extinctive (comma) selection are included, as performance was found to be superior to the elitist (plus) selection strategy. It has been widely accepted that the extinctive selection mechanism is most appropriate when a self-adaptive mutation operator is adopted (Schwefel 1995). As in previous experimentation, selection pressure is maintained at a constant ratio of $\frac{\mu}{\lambda} = \frac{1}{7}$, with exact figures indicated for each run. The objective of each algorithm is to minimise the relative spectral error. Population sizes for the multimembered ESs are set to $\mu = 200$ and $\lambda = 1{,}400$ and the number of clusters within the CES is set to

40. Values for each synthesis parameter are represented as a floating-point approximation of a real number.

9.1 Static contrived tone matching

In this first group of experiments, each EA is applied to match 30 randomly generated contrived target tones for each of the three simple FM synthesis models depicted in Fig. 3. Each contrived target tone is synthesised by drawing parameter values uniformly at random from within the object range of each synthesis parameter. Results are tabulated indicating the population sizes in the standard ES notation and the recombination type (intermediate (int), centroid (cen) or discrete (dis)). Algorithmic performance is measured by the average error of the fittest solutions evolved in each run and the number of runs classed to be *successful*, where a successful match produces a relative spectrum error of less than 0.01.

The number of synthesis parameters (the problem dimensionality) is 4, 8 and 12 for the single, double and triple FM synthesis models, respectively. The parameter range for each simple FM element is indicated in Table 1. The values for f_c and f_m are multiplied by 440 Hz (concert pitch) to give a frequency.

From the results provided in Table 2, it is apparent that the problem space becomes less tractable as the number of parallel simple FM elements in the matching synthesiser is increased. This result is expected, as all algorithmic parameters remain constant while the dimensionality of the search space increases. The CES with discrete recombination produces the greatest number of successful matches; however, none of the tested EAs are able to produce successful matches when optimising parameters for the triple simple FM model. The CES with discrete recombination also produces the smallest average error, with the exception of the ES with discrete recombination in application to the triple simple FM model and the CES with centroid recombination in application to the single simple FM

Table 1 Static synthesis parameter summary

Parameter	Range
f_c, f_m	0.0–8.0
A	0.0–1.0
I	0.0–8.0

Table 2 Static contrived tone matching results

Algorithm	Recomb	Success	Mean error (σ)
Single simple FM			
ES (200, 1400)	dis	13	0.25 (0.26)*
ES (200, 1400)	int	6	0.46 (0.27)*
MSES (1 + 1) × 1400	–	0	0.16 (0.11)*
MSES (1 + 1) × 350	–	6	0.12 (0.13)*
MSES (1 + 1) × 175	–	8	0.10 (0.12)*
MSES (1 + 1) × 100	–	9	0.15 (0.18)*
CES (200, 1400)	dis	29	0.00 (0.01)
CES (200,1400)	cen	19	0.05 (0.12)
Double simple FM			
ES (200, 1400)	dis	2	0.32 (0.15)*
ES (200, 1400)	int	0	0.50 (0.18)*
MSES (1 + 1) × 1400	–	0	0.38 (0.10)*
MSES (1 + 1) × 350	–	0	0.38 (0.13)*
MSES (1 + 1) × 175	–	0	0.41 (0.15)*
MSES (1 + 1) × 100	–	0	0.40 (0.13)*
CES (200, 1400)	dis	3	0.20 (0.10)
CES (200, 1400)	cen	3	0.31 (0.14)*
Triple simple FM			
ES (200, 1400)	dis	0	0.28 (0.14)
ES (200, 1400)	int	0	0.45 (0.14)*
MSES (1 + 1) × 1400	–	0	0.41 (0.11)*
MSES (1 + 1) × 350	–	0	0.39 (0.12)*
MSES (1 + 1) × 175	–	0	0.40 (0.10)*
MSES (1 + 1) × 100	–	0	0.43 (0.15)*
CES (200, 1400)	dis	0	0.27 (0.08)
CES (200, 1400)	cen	0	0.36 (0.10)*

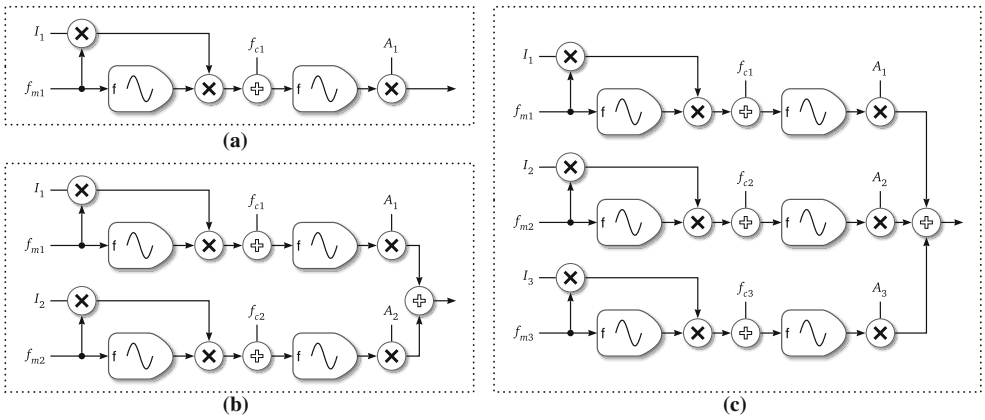

Fig. 3 **a** Single, **b** double and **c** triple parallel static simple FM arrangements

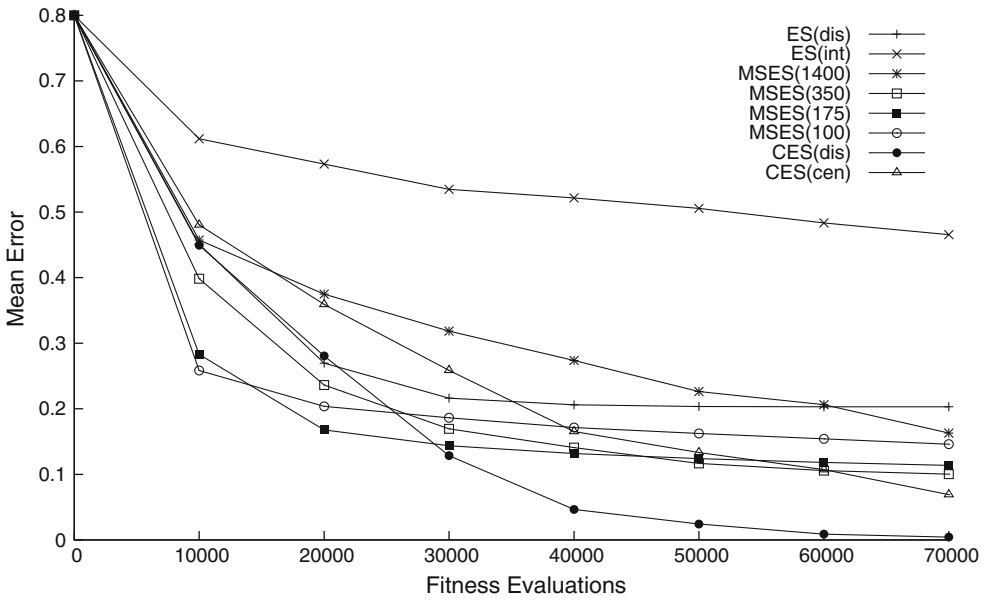

Fig. 4 Simple FM contrived matching convergence

Table 3 Static acoustic target fundamental frequencies

Instrument	Pitch	Frequency
Oboe	G5	783.99
Trumpet	C5	523.25
Muted trumpet	F5	698.46

model, where the observed reduction in error was not statistically significant. The statistical significance of the measured improvement of the CES with discrete recombination with respect to the other algorithms tested is indicated in the table with an asterisk (*). The results indicate that the CES is very effective at navigating the search space of the single simple FM synthesiser; this performance advantage is attributed to the improved maintenance of population species resulting from the inclusion of *k*-means cluster analysis and the associated selection and recombination operators.

9.2 Convergence plot

The convergence characteristics of the tested algorithms are shown in Fig. 4. Plots indicate the average convergence over all 30 matches from the above experimentation with the simple FM synthesiser. The MSESs with the smallest number of search points shows the fastest initial progress, while the CES converges towards the smallest error.

9.3 Static acoustic tone matching

The experimentation performed above is now repeated for the CES with the same population and cluster sizes, substituting the synthetic contrived tones with real tones extracted from the sustain (relatively stable, middle section) of three acoustic instrument tones. The target tones originate from an oboe, trumpet and muted trumpet, recorded and produced by Opolko and Wapnick (1989). Details of the three tones are provided in Table 3.

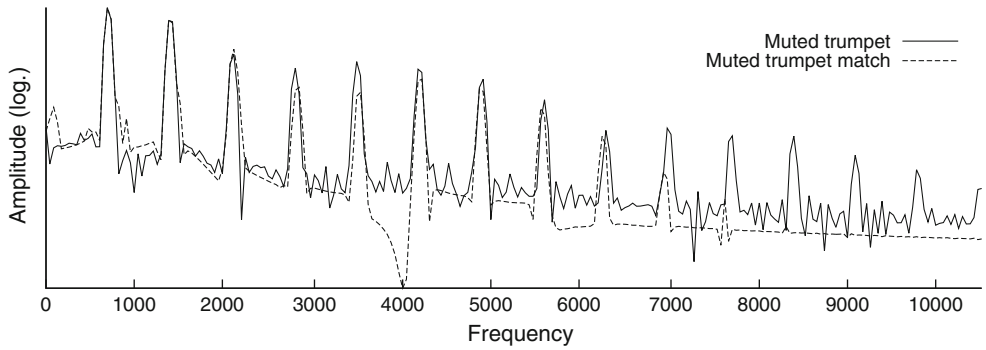

Fig. 5 Muted trumpet tone with match

Table 4 Static acoustic matching results

Algorithm	Synthesiser	Oboe mean error (σ)	Trumpet mean error (σ)	Muted trumpet mean error (σ)
CES	Single	0.22 (0.00)	0.20 (0.03)	0.18 (0.01)
CES	Double	0.13 (0.03)	0.14 (0.03)	0.15 (0.02)
CES	Triple	0.10 (0.02)	0.14 (0.02)	0.11 (0.02)

Table 5 Dynamic synthesis envelope parameter summary

Parameter	Range
a_c, a_m	0.0–1.0
d_c, d_m	0.0–1.0
s_c, s_m	0.0–1.0
r_c, r_m	0.0–1.0

Figure 5 shows the spectra of the muted trumpet target and matched spectra overlaid on a log. amplitude scale synthesised by the triple simple FM model. All high-amplitude partials are accurately represented in the match, with only minor differences and omissions in the lower amplitude high-frequency components.

Interestingly, the results (Table 4) largely exhibit the opposite trend to those produced in the contrived matching experiments. Previously, the relative error rates were shown to increase when using the larger synthesis models, whereas here, the error rates decrease with the larger model, with the exception of the trumpet tone match with the triple FM synthesiser. These results illustrate the opposing limitations of the matching process. The CES is well suited to the problem domain of the single simple FM model. The small standard deviation for this model suggests that the majority of the runs have converged at the same solution, the optimum for this synthesis model. In attempting to match the trumpet target tone the CES has reached the limitations of the matching synthesiser. This error result cannot be improved unless a more sophisticated synthesis model is employed. The introduction of additional parallel simple FM elements to the model results directly in a more accurate match. While the earlier results suggested that the CES is less effective at exploring the double and triple simple FM synthesis spaces, when approaching the limitations of the matching synthesiser, the larger space is beneficial.

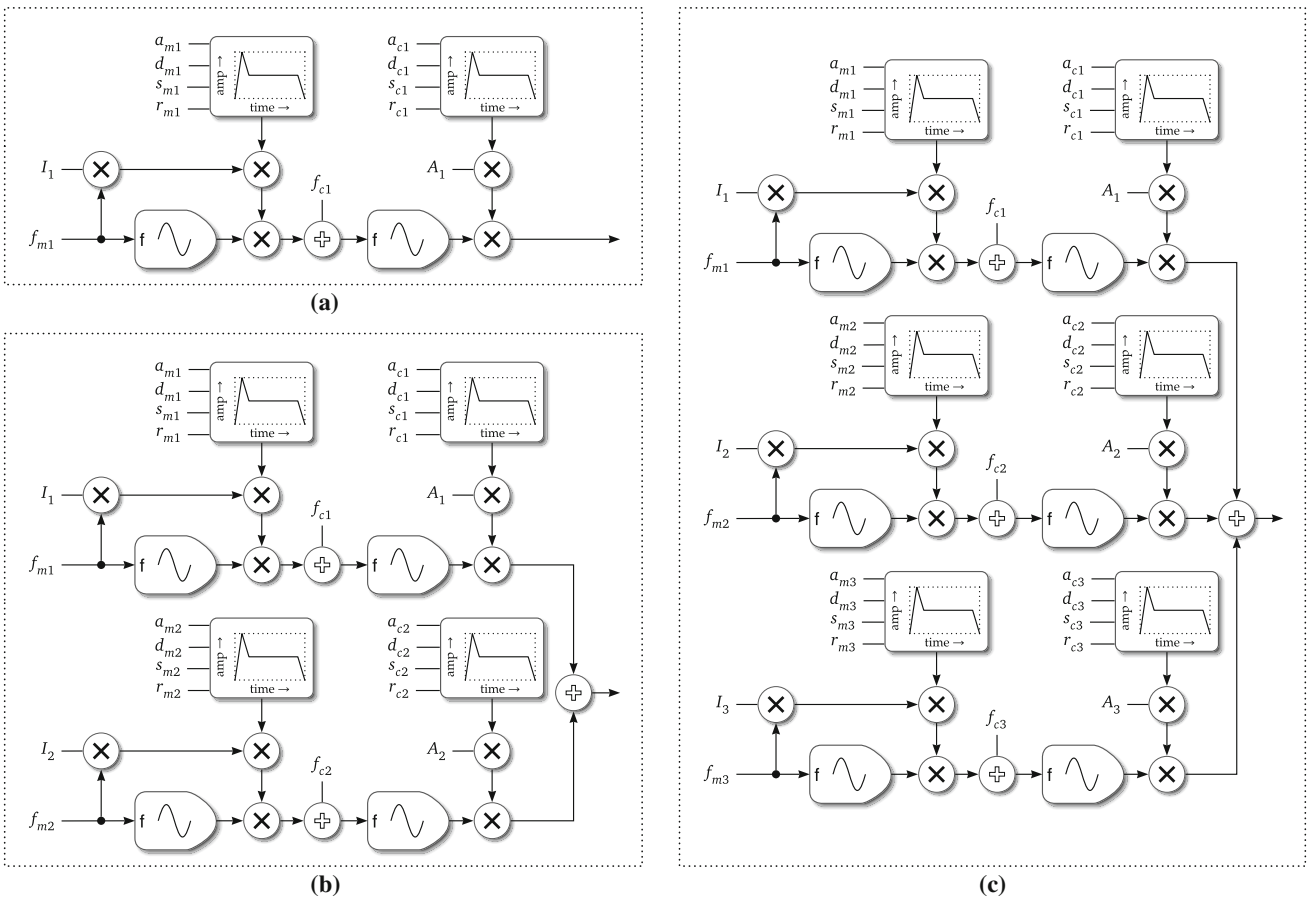

Fig. 6 **a** Single, **b** double and **c** triple parallel dynamic simple FM arrangements

Table 6 Dynamic contrived sound matching results

Algorithm	Recomb	Success	Mean error (σ)
Single simple FM			
ES (200, 1400)	dis	11	0.03 (0.03)
ES (200, 1400)	int	0	0.27 (0.15)*
MSES (1 + 1) × 1400	–	0	0.15 (0.08)*
MSES (1 + 1) × 350	–	0	0.13 (0.08)*
MSES (1 + 1) × 175	–	0	0.13 (0.08)*
MSES (1 + 1) × 100	–	0	0.13 (0.09)*
CES (200, 1400)	dis	16	0.02 (0.02)
CES (200, 1400)	cen	3	0.08 (0.03)*
Double simple FM			
ES (200, 1400)	dis	0	0.08 (0.07)
ES (200, 1400)	int	0	0.29 (0.14)*
MSES (1 + 1) × 1400	–	0	0.22 (0.10)*
MSES (1 + 1) × 350	–	0	0.18 (0.10)*
MSES (1 + 1) × 175	–	0	0.18 (0.10)*
MSES (1 + 1) × 100	–	0	0.18 (0.11)*
CES (200, 1400)	dis	0	0.06 (0.05)
CES (200, 1400)	cen	0	0.10 (0.10)
Triple simple FM			
ES (200, 1400)	dis	0	0.14 (0.08)
ES (200, 1400)	int	0	0.32 (0.10)*
MSES (1 + 1) × 1400	–	0	0.27 (0.09)*
MSES (1 + 1) × 350	–	0	0.24 (0.08)*
MSES (1 + 1) × 175	–	0	0.24 (0.08)*
MSES (1 + 1) × 100	–	0	0.25 (0.09)*
CES (200,1400)	dis	0	0.11 (0.06)
CES (200,1400)	cen	0	0.20 (0.07)

Table 7 Dynamic acoustic target fundamental frequencies

Instrument	Pitch	Frequency
Muted French horn	D5	587.33
Trumpet	F5	698.46
Oboe	F♯5	739.99

Table 8 Dynamic acoustic matching results

Algorithm	Synthesiser	Oboe mean error (σ)	Trumpet mean error (σ)	French Horn mean error (σ)
CES	Single	0.15 (0.00)	0.20 (0.02)	0.16 (0.01)
CES	Double	0.10 (0.01)	0.13 (0.01)	0.11 (0.01)
CES	Triple	0.10 (0.01)	0.13 (0.01)	0.11 (0.01)

9.4 Dynamic contrived sound matching

In the next group of experiments, each EA is applied to match 30 randomly generated contrived target sounds, using each of the three simple FM synthesis models depicted in Fig. 6. The model represents the most fundamental time-varying simple FM synthesis structure as defined originally by Chowing (1973). In the time-varying synthesis models the carrier amplitude A and modulation index I are controlled by simple envelope generators. Each envelope generator introduces four parameters: attack (a), decay (d), sustain (s) and release (r), which enable the envelope-modulated parameters to change over time. This temporal control results in the production of time-varying sound textures and has been used within commercial FM synthesisers.

As with the static case, each contrived target sound is synthesised by drawing parameter values uniformly at random from within the object range of each synthesis parameter. Results are again tabulated indicating the population sizes in the standard ES notation with the recombination type employed: intermediate (int), centroid (cen) or discrete (dis). Algorithmic performance is measured by the average error of the fittest solutions evolved in each run and the number of runs classed to be *successful*, where a successful match produces a relative spectrum error of less than 0.01.

The number of synthesis parameters (the problem dimensionality) for the dynamic models are 12, 24 and 36 for the single, double and triple simple FM synthesis models respectively. The parameter range for each simple FM form is as shown in Table 1 with the envelope parameters indicated in Table 5. The value of a is mapped to 0–50 % of the target sound duration, d and r are mapped to 0–25 % of the target sound duration and the value of s is used directly as a multiplier for the sustain period. The sustain period is adjusted automatically to ensure that candidate and target sound durations are equal.

As with the static case it is apparent that the problem space becomes less tractable as the number of parallel simple FM elements in the matching synthesiser is increased. The CES with discrete recombination is again most consistent in the production of small error values and the greatest number of successful matches. However, the performance difference between both CES types and the ES with discrete recombination is not statistically significant in all cases. Again, the statistical significance of the measured improvement of the CES with discrete recombination with respect to the other algorithms tested is indicated in the table with an asterisk (*). The ES with discrete recombination produces smaller errors this time than with the equivalent static tone problem tested earlier. As the ES is more susceptible to becoming trapped at local optima than the niching-based algorithms, this result could suggest that the time-varying FM search space is more tractable than the equivalent static tone space. However, the consistently poor performance of the MSES algorithms,

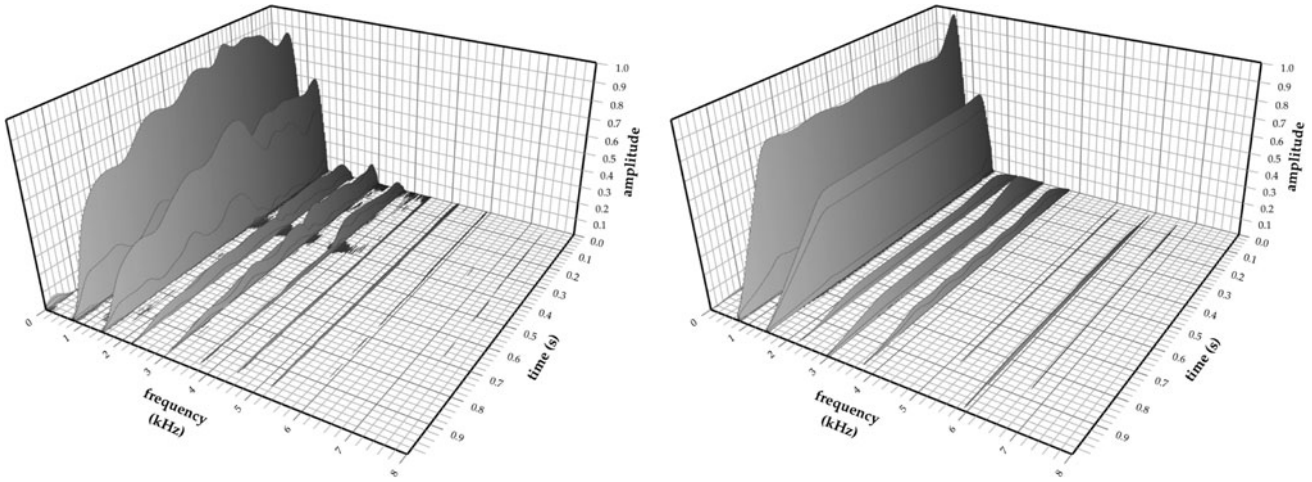

Fig. 7 Dynamic oboe sound target (*left*) and match (*right*)

combined with the convergence of each CES cluster to an independent niche (not shown), suggests that there are still many potential matches for each target sound, and thus the matching space is extensively multimodal. These results suggest that the time domain control introduces search space characteristics that are beneficial to the EAs; this may be a result of the averaging of the error throughout the time domain of the target/candidate sound or due to the introduction of envelope parameters in the matching synthesisers. Overall, every EA struggles to produce successful matches when optimising parameters for the triple simple FM model. The contrived matching method enables this deficiency to be attributed to the optimisation algorithms, since the target sound is known to exist within the search space (Table 6).

9.5 Dynamic acoustic tone matching

In this section the CES derivatives of the time-varying FM-matching system with the same population and cluster sizes are used to optimise matches to dynamic acoustic sounds. The target set is comprised of three instrument samples, again produced by Opolko and Wapnick (1989): muted French horn, trumpet and oboe. Details of each sound are provided in Table 7. All EAs optimise a match for each target sound using the dynamic FM synthesis models depicted in Fig. 6. The mean relative spectral error of the best individual for each test case is provided in Table 8; results are computed from the average error of 30 independent and randomly initialised runs.

The reduction in error observed between the matches produced on the single and double simple FM models does not extend further when matches were performed using the triple FM model, where there is no improvement in accuracy when the most complicated synthesiser is tested. The larger triple FM model would certainly be capable of producing more exact matches than the smaller models, but the CES is unable to exploit this advantage in these tests. This is due to the maintenance of fixed population sizes while the search space dimensionality is increased. This hypothesis can be tested by repeating the experiment with a doubling of the population size, cluster quantity and the number of generations for which each algorithm runs. In matching the oboe sound with the triple FM model the error was shown to drop to 0.08 (0.01). The accuracy of this particular match can be compared visually by observing time/frequency plots shown in Fig. 7. The plots illustrate that all of the significant partials visible in the target sound are well represented in the match, with the exception of a low-amplitude partial positioned at 4,440 Hz, which is absent in the match. The plot also highlights the limits of the synthesiser envelope generators, which are unable to reproduce the subtle fluctuations in the partial amplitudes.

10 Conclusions

This paper discussed prior research activity in automated evolutionary sound matching and, in doing so, the components of a sound matching system that contribute to match inaccuracy were introduced. Considerations were made for how match accuracy can be improved and two approaches were identified: simplifying the problem space and improving the optimiser. The subsequent sections of the paper focused on the latter and a CES, designed to concurrently optimise multiple distinct solutions within a single population, was developed. This work represents the earliest applications of the evolution strategy and niching evolutionary optimisers to the problem of evolutionary sound matching. The presented algorithm was then applied

with a fixed population to optimise static and dynamic sounds using different configurations of the parallel simple FM arrangement and compared with conventional EAs. A contrived matching method enabling the quantitive comparison between optimisation algorithms was adopted enabling match inaccuracies to be attributed directly to the performance of the optimisers. The results of this comparison indicated that the CES was more consistent in the production of smaller errors and more accurate sound matches than conventional and hill-climbing ESs.

Much work is still to be done in this domain. For example, experimentation with different optimisation algorithms might result in improvements in both match accuracy and time. Alternative sound similarity measures and synthesisers might provide a less complicated view of the problem space in which experimentation with predictive measures of problem difficulty might prove useful (Naudts and Kallel 2000).

Acknowledgments The author would like to thank Professor Larry Bull, Dr David Creasey and Professor Tony Pipe at the University of the West of England for their support and encouragement in the preparation of this work.

References

Ashley R (1986) A knowledge-based approach to assistance in timbral design. In: Proceedings of the 1986 International Computer Music Conference. Royal Conservatory, Den Haag, The Netherlands, pp 11–16

Beauchamp J, Horner A (2003) Error metrics for predicting discrimination of original and spectrally altered musical instrument sounds. J Acoust Soc Am 114(4):2325

Beyer HG (2001) The theory of evolution strategies. Springer

Bozkurt B, Yüksel KA (2011) Parallel evolutionary optimization of digital sound synthesis parameters. In: Proceedings of the 2011 international conference on applications of evolutionary computation. Springer, Berlin

Chowning JM (1973) The synthesis of complex audio spectra by means of frequency modulation. J Audio Eng Soc 21(7):526–534

Das S, Suganthan PN (2011) Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Technical Report

Delprat N (1997) Global frequency modulation laws extraction from the gabor transform of a signal: a first study of the interacting component case. IEEE Trans Speech Audio Process 5(1):64–71

Garcia R (2002) Automatic design of sound synthesis techniques by means of genetic programming. In: Proceedings of the 113th convention of the Audio Engineering Society, Preprint 5654. Los Angeles, CA

Gonzalez TF (1985) Clustering to minimize the maximum intercluster distance. Theor Comput Sci 38(2–3):293–306

Holland J (1975) Adaptation in natural and artificial systems. University Press, Ann Arbor

Horner A (1997) A comparison of wavetable and FM parameter spaces. Comput Music J 21(4):55–85

Horner A (1998) Nested modulator and feedback FM matching of instrument tones. IEEE Trans Speech Audio Process 6(6):398–409

Horner A, Beauchamp J, Haken L (1993) Machine tongues xvi: genetic algorithms and their application to FM, matching synthesis. Comput Music J 17(4):17–29

Horner A, Beauchamp J, Haken L (1993) Methods for multiple wavetable synthesis of musical instrument tones. J Audio Eng Soc 41(5):336–356

Horner A, Beauchamp J (2006) Error metrics to predict discrimination of original and spectrally altered musical instrument sounds. J Audio Eng Soc 54(3):140–156

Justice JH (1979) Analytic signal processing in music computation. IEEE Trans Acoust Speech Signal Process 27(6):670–684

Lim SM, Tan BTG (1999) Performance of the genetic annealing algorithm in DFM synthesis of dynamic musical sound samples. J Audio Eng Soc 47(5):339–354

MacQueen J (1967) Some methods for the classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability

Mahfoud SW (1995) Niching methods for genetic algorithms. PhD thesis, Urbana, IL, USA

McDermott J, Griffith NJL, O'Neill M (2005) Toward user-directed evolution of sound synthesis parameters. In: EvoWorkshops. Springer

McDermott J, Griffith NJL, O'Neill M (2008) Evolutionary computation applied to sound synthesis. In: The art of artificial evolution. Springer

Miranda ER (2002) Computer sound design: synthesis techniques and programming. 2nd edn, Focal Press, Oxford

Mitchell T, Creasey D (2007) Evolutionary sound matching: A test methodology and comparative study. In: Proceedings of the sixth international conference on machine learning and applications. IEEE

Mitchell T, Pipe AG (2006) A comparison of evolution strategy-based methods for frequency modulated musical tone timbre matching. In: Proceedings of the seventh international conference on adaptive computing in design and manufacture. Bristol

Mitchell TJ (2010) An exploration of evolutionary computation applied to frequency modulation audio synthesis parameter optimisation. PhD thesis, Univeristy of the West of England, Bristol

Naudts B, Kallel L (2000) A comparison of predictive measures of problem difficulty in evolutionary algorithms. IEEE Trans Evol Comput 15(4):1–15

Opolko F, Wapnick J (1989) McGill University Master Samples (MUMS). 11 CD-ROM set, Faculty of Music, McGill University, Montreal, Canada

Ostermeier A, Gawelczyk A, Hansen N (1994) A derandomized approach to self-adaptation of evolution strategies. Evolutionary Computation 2(4):369–380

Payne R (1987) Microcomputer based analysis/ resynthesis scheme for processing sampled sounds using fm. In: Proceedings of the international computer music conference

Rechenberg I (1965) Cybernetic solution path of an experimental problem. Technical report, RAE Translation 1122, Farnborough, Hants

Riionheimo J, Välimäki V (2003) Parameter estimation of a plucked string synthesis model using a genetic algorithm with perceptual fitness calculation. EURASIP J Appl Signal Process 2003(8): 791–805

Risberg JS (1980) Non-linear estimation of fm synthesis parameters. In: Proceedings of the 67th Convention of the Audio Engineering Society, 1685. New York

Roads C (1996) The computer music tutorial. MIT Press, Cambridge

Schottstaedt B (1977) The simulation of natural instrument tones using frequency modulation with a complex modulating wave. Comput Music J 1(4):46–50

Schwefel HP (1995) Evolution and optimum seeking. Wiley, USA

Streichert F, Stein G, Ulmer H, Zell A (2000) A clustering based niching ea for multimodal search spaces. In: Proceedings of the International Conference Evolution Artificielle. Springer

Wehn K (1998) Using ideas from natural selection to evolve synthesized sounds. In: Proceedings of the digital audio effects DAFX98. Barcelona, pp 159–167

Yee-King M, Roth M (2011) A comparison of parametric optimization techniques for musical instrument tone matching. In: Proceedings of the 130th Convention of the Audio Engineering Society

Yee-King MJ, Roth M (2008) Synthbot - an unsupervised software synthesizer programmer. In: Proceedings of the International Computer Music Conference ICMC08. Belfast, N. Ireland