Analytical Tools for Group Additive Synthesis

Barry Eaglestone,
Department of Computing,
University of Bradford,
BRADFORD,
West Yorkshire BD7 1DP
burry@uk.ac.bradford.d.computing

Shaun Oates,
Microcomputer Music Research Unit,
Department of Computing,
University of Bradford,
BRADFORD West Yorkshire BD7 1DP
shaun@uk.ac.bradford.computing

ABSTRACT: Group additive synthesis (GAS), a compact representation of sound, consists of component partials as members of groups with common frequency and amplitude envelopes. Resynthesis is achieved by applying amplitude scaling factors to the group envelope. GAS reduces the high overheads of additive synthesis, but algorithmic determination of an optimum GAS configuration is NP-complete and therefore intractable. We present a technique which performs an effective, if not optimum, application of GAS, and show this to be an improvement on previous methods. The paper identifies further research areas for GAS and suggests additional applications.

1 Introduction

Real-time additive synthesis is useful for precise definition and control of sounds but at a cost. Its considerable storage, oscillator, and computational power overheads [1] makes the use of data reduction techniques e.g., [2-4] desirable.

One such method, namely Group Additive Synthesis (GAS) [2,5], was first formalised by Kaczkowski. In this method the component partials of a sound are divided into groups, each group having common amplitude and frequency envelopes. On resynthesis these envelopes are scaled to the appropriate magnitude and used to generate partials which approximate the originals. This can provide large compression ratios and reduce the amount of synthesis hardware used by a real-time system whilst retaining much of the perceptual identity of the sound.

This paper examines GAS, concentrating on the process used to provide the grouping arrangement. The problem of grouping partials has been found to be, in theory, computationally intractable. However, satisfactory non-optimal solutions can be derived algorithmically, and we present an improved version of the grouping process based on techniques developed from the field of classification [6].

2 Application of GAS for Sound Simulation

GAS is a destructive compression technique based on the hypothesis that many of the small differences between broadly similar envelope shapes of a sound's partials are not perceptually significant. The reconstruction of a GAS sound can be done in either real-time or non-real-time. The Bradford Musical Instrument Simulator (BMIS) [5] is an example of an instrument which generates sound in real-time from the GAS representation.

Quality simulations of musical instruments have been achieved on BMIS through intuitive application of GAS; partial groupings and envelopes have been determined by visual inspection of FFT and phase vocoder analyses and by trial and error. However, an analytical tool for applying GAS potentially offers a more quality-sourced and productive basis for resynthesis of sounds. Such a tool must be based on: (i) a metric for quantifying the similarity between envelopes; (ii) an algorithm for grouping partials with similar envelopes; (iii) some function for deriving an envelope for a partial group.

3 Similarity Metric

The first stage in deriving a grouping structure is to quantify the 'difference' between partials. An ideal metric would be one in which partial dissimilarity is measured by the comparison of various perceptual features (percepts). However the current models of timbral perception do not allow quantitative comparison of percepts between partials. Therefore both we and Kaczkowski quantify the dissimilarity between partials as a simple Euclidean measure of area difference between partial envelope shapes. However it is known that this technique does not allocate the correct importance to features which are arithmetically small but perceptually significant, such as amplitude overshoot and other transients [7], and so this area requires further research.

4 Group Calculation

The grouping algorithm must allocate each partial to a group on the basis of the derived partial differences, with the objective of minimising the dissimilarity between partials in a group. This can be expressed more formally, e.g. graph theory, as the minimal k-partition of a weighted graph.
the number of groups is k and the dissimilarity measurements form a weighted graph. However, Bruckner [8] has shown this problem to be NP-complete. The set of NP-complete problems is such that the existence of a polynomial time-complexity solution algorithm for one of them, proves the existence of such an algorithm for them all. However, as yet no such algorithms are known. Execution time therefore becomes prohibitive for all but trivial instances of the problems. For further exploration of NP-completeness see [9].

Consequently, the best we can hope for is an heuristic-based algorithm which will produce a 'good' but not necessarily optimum grouping.

5 Envelope Calculation

The envelope should be calculated so it best represents the group as a whole. Again the ideal method of envelope computation would be percept-based, but we confine our attention to a purely geometric computation.

6 The Method of Group Calculation Used

The problem of grouping items according to their similarity has been extensively covered by researchers in the field of classification. The grouping method we decided upon is one of the more commonly used classification techniques.

The particular scheme used is a hierarchical clustering scheme (HCS) [11] which operates as follows:

Step 1: Each element to be clustered is placed in a one member group; these form a list of "free" groups.

Step 2: A new group is created by locating and merging the two most similar groups in the free list. Merging is done by creating a new set of dissimilarity relationships between this group and all the other groups. The set is calculated as the average dissimilarity of the two sub-groups, and the new group size is the difference between the two sub-groups. The two sub-groups are removed from the free list and are added as children of the new group in the manner of a binary tree.

Step 2 is repeated until there is only one free group left which encompasses all of the original partials.

The algorithm results in a tree with the following properties: (i) each node represents a group; (ii) each branch indicates membership of that group; (iii) the leaves of the tree represent the component partials of a sound; (iv) the root of the tree represents a group which has all the component partials as members; (v) the nodes of the tree have an associated group size value which represents the average dissimilarity of the members of the group.

The tree structure provides a range of possible grouping systems ranging from many groups (low-distortion/low-compression) near the leaves, to few groups (high-distortion/high-compression) at the root. After group selection, the tree is cut into a number of distinct subtrees, each representing a group; the leaf nodes of each subtree represent the component partials.

The main tree holds information on distortion introduced at different grouping levels, and so may be parsed to extract data on the most desirable of these. Each node represents two values: a group count, and group size. The tree can be parsed to provide a function of distortion introduced versus number of groups, which can be manipulated into a function which gives error-compression ratio vs number of groups.

If the error-compression ratio vs number of groups function is displayed graphically, saddle points in the curve become apparent. These positions represent what Knuskai [12] has termed 'natural' grouping systems, i.e. groups which are inherent in the data, rather than an arbitrary imposition.

In terms of the GAS technique these positions also represent the grouping combinations where the tradeoff between compression and error is most favourable.

7 Results and Conclusions

This technique improves on the algorithm originally used by Kleczkowski in three areas: (i) HCS provides a range of possible groupings and so can form the basis for an interactive see-and-audit method of sound reduction; (ii) Empirical testing indicates a reduction of error in the order of 5-10%; (iii) HCS has an easier user interface, in that it provides readily comprehensible information, and requires less complex input.

In addition, we have established the connection of GAS and the well researched area of classification studies. This makes available tools for further reasoning about GAS application.

8 Proposals for further work

The perceptual areas of the technique have known faults (see 3 and 5). Principally a method is needed of assessing the perceptual similarity of envelope shapes, perhaps using rule-induction systems and neural nets.
The process of clustering opens up possibilities for other forms of compact additive synthesis sound representations, for example instead of clustering partial envelopes according to similarity, spectral envelopes could be grouped to achieve a compact form of time interpolation [13].

Experiments have been performed in non-real-time on single instrument sounds. However, if the system is to be extended to cover real-time simulations of the whole of an instrument, several issues concerning the constraints imposed by the real-time hardware must be addressed.

9 Summary

GAS reduces the high overheads of additive synthesis, but algorithmic determination of optimum GAS is NP-complete and therefore intractable. In this paper we have presented a heuristically-based HCS algorithm which performs an effective, if not optimum, application of GAS. We have shown this algorithm to be an improvement on the previous one of Kiczales. Further research is necessary to take into account the perceptual aspects of a sound, and the implications of instrument simulation in real-time.

References