Abstract. This paper presents a method for representing trees using constraint logic programming over finite domains. We describe a class of trees that is of particular interest to us and how we can represent the set of trees belonging to that class using constraints. The method enables the specification of a set of trees without having to generate all of the members of the set. This allows us to reason about sets of trees that would normally be too large to use. We present this research in the context of a system to generate expressive musical performances and, in particular, how this method can be used to represent musical structure.

1 Introduction

This paper describes how constraints can be used to represent a specific class of trees that have the following properties:

Rooted - each tree has a node distinguished as the root node.

Ordered - the children of each node are distinct and cannot be re-ordered without changing what the tree represents.

Constant Depth - the leaf nodes of each tree are all the same distance from the root.

Strict - at each depth, one of the nodes has at least two successors.

The number of distinct trees in this class is large for each \( n \), where \( n \) is the number of leaf nodes. If \( n \geq 10 \) the set of trees described can not easily be manipulated or used within a computer system. We present here an efficient way of representing this large set of trees, using constraint logic programming, that enables us to use this class of trees in our research.

The structure of the paper is as follows. The next section explains why we are interested in representing sets of trees in the context of music. We then present some implementation details including our representation and the constraints used to specify the trees of interest. Some results are presented that illustrate the effectiveness of this method. Finally, we end with our conclusions.

* Ben Curry is supported by UK EPSRC postgraduate studentship 97305827
2 Motivation: Grouping Structure

This work forms part of our research into creating an expressive musical performer that is capable of performing a piece of music alongside a human musician in an expressive manner.

An expressive performance is one in which the performer introduces variations in the timing and dynamics of the piece in order to emphasise certain aspects of it. Our hypothesis is that there is a direct correlation between these expressive gestures and the musical structure of the piece and we can use this link to generate expressive performances.

The theory of musical structure we are using is the Generative Theory of Tonal Music (GTMM) by Lerdahl and Jackendoff (1983). The theory is divided into four sections that deal with different aspects of the piece’s musical structure.

We are particularly interested in the grouping structure which corresponds with how we segment a piece of music, as we are listening to it, into a hierarchy of groups. It is this hierarchy of groups that we seek to represent with our trees.

The rules are divided into two types: well-formedness rules that specify what structures are possible; and preference rules that select, from the set of all possible structures, those that correspond most closely to the score.

The rules defining grouping structures are based on principles of change and difference. Figure 1 shows four places where a grouping boundary may be detected (denoted by a ‘*’). The first case is due to a relatively large leap in pitch between the third and fourth notes in comparison to the pitch leaps between the other notes. The second boundary occurs because there is a change in dynamics from piano to forte. The third and fourth boundaries are due to changes in articulation and duration respectively.

![Fig. 1. Points in the score where grouping rules may apply](image)

Figure 2 shows an example of a grouping structure for a small excerpt of music. We can see that the music has been segmented into five different groups, one for each collection of three notes. The musical rest between the third and fourth groups causes a higher level grouping boundary that makes two higher level groups which contain the five groups. These groups are then contained within one large group at the highest level.

The grouping structure can be represented with a tree. Figure 3 shows a tree representation (inverted, to aid comparison) for the grouping structure shown in Fig. 2. The leaf nodes at the top of the tree correspond to the notes in the score, and the branches convey how the notes are grouped together. This is an