A prefix tree-based user interface for exploring abstract structure in Irish folk tunes

Kieran Jayes, Seán Doherty, and James McDermott*

1 Introduction

This paper briefly introduces a user interface for exploration of the abstract structure of a collection of folk tunes. The dataset is *The Dance Music of Ireland: 1001 Gems* collected by O'Neill (1907) (henceforth *Gems*). It is a very central resource for Irish traditional musicians, even known as "The Book" or "The Bible" (Doherty, 2022).

Some aspects of high-level musical structure are quite homogeneous in the dataset. For example, each tune typically contains two or more major parts, and a part is typically 8 bars in length, with a clear-cut transition between parts.

Our focus is on structure at a level lower than parts, but above notes: we are concerned with repetition and variation at the bar level. With this structure we can identify tunes which share common abstract structure even while their lower-level contents – raw notes – differ.

2 Bar-Level Analysis

In previous work (Doherty, 2022), one of the authors (SD) has analysed the bar-to-bar melodic structure of the 365 double jigs in *Gems*, justifying this level of analysis in musicological terms. (A double jig is a dance tune in 6/8 with an emphasis on eighth notes rather than mixed quarter and eighth notes.) This section draws on that work.

The system of analysis works by assigning a letter to each bar, starting with *a* for the first bar and ignoring pick-up bars. For subsequent bars:

- If the bar is almost identical (5 or 6 of the 6 notes identical) to a previous bar, it is assigned the same letter as the previous bar. For example in Fig. 1, bars 1 and 5 are notated as *a*.
- If the bar is similar to a previous bar (3-4 notes identical, or a direct transposition), it is assigned as a variation of the previous bar, with the same letter but an added numeral, e.g. in Fig. 1, bar 3 is a variation of bar 1 and so is assigned the symbol a^1.
- If the bar is more dissimilar than this, it is assigned a new letter in sequential order, e.g. bar 2 is different from bar 1 and so is named *b*.
- If the bar is in part *B* or later and is similar to a bar in a previous part, the previous part name is used as a prefix, e.g. bar 15 is labelled *Ad* indicating bar *d* in part *A*.

Although only the double jigs were described in previous work, all 1001 *Gems* have been annotated manually and are used in our user interface, described next.

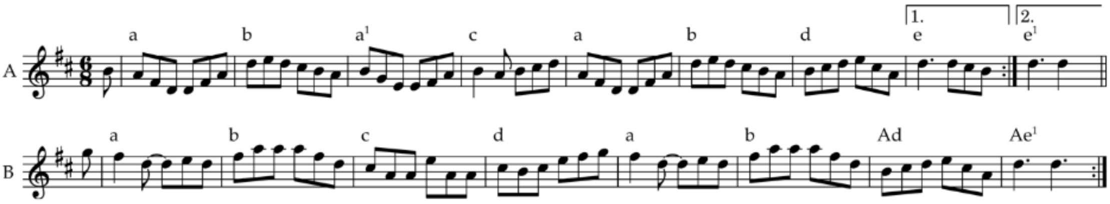


Figure 1: Shandon Bells, *Gems #1*. There are two parts, *A* and *B*, each composed of 8 bars. In our structural analysis, distinct bars are labelled with distinct lowercase letters, with letter-number symbols to indicate variants. From Doherty (2022).

*Corresponding author. james.mcdermott@universityofgalway.ie. This work is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004746 (Polifonia: a digital harmoniser for musical heritage knowledge, H2020-SC6-TRANSFORMATIONS).

3 Our User Interface

We have implemented a user interface (UI) allowing the user to explore the abstract structures of the *Gems*. The UI is based on a *prefix tree*: a tree data structure which takes advantage of common prefixes in data. For each sub-tree, the root represents the common prefix of all items within that sub-tree. Our 1001 structures can be naturally represented by a prefix tree. All begin with a and so the root of the tree is labelled a, as shown in Fig. 2.

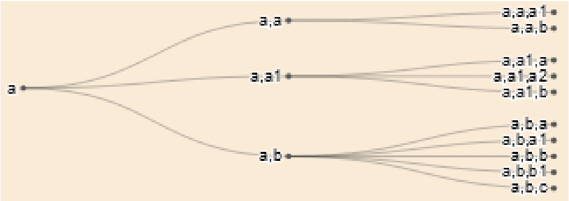


Figure 2: The first three levels of the prefix tree visualisation. All 1001 structures begin with a, so it becomes the root node. It has three children representing the three possible continuations a, a^1, and b. Already at depth 3 we see that some possible structures do not occur: aaa is not present.

In order to save space, a prefix tree stores a string such as $abcd$ using 4 nodes as $a - b - c - d$. For ease of use we visualise it as $a - ab - abc - abcd$.

Each leaf of the prefix tree represents a single tune. However, as shown in Fig. 3, a single structure may be shared by many tunes, or may be unique to a single tune. Two groups of tunes which are nearby in the UI will generally have similar structures.

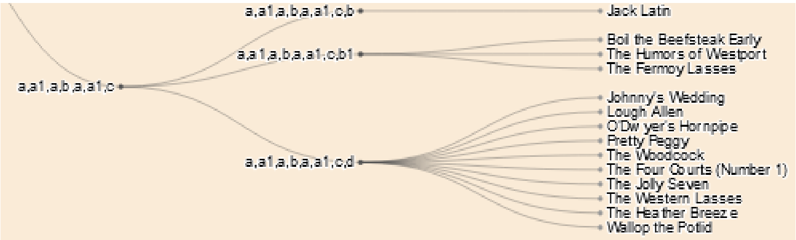
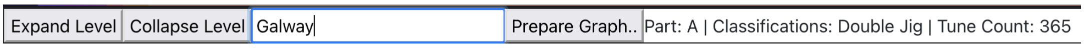


Figure 3: Leaves represent individual tunes. Some structures are present in many tunes. Clicking on an individual tune title (right of figure) shows the tune in standard notation (from *thesession.org*) with a button to render the tune via MIDI.

The UI allows the user to easily explore rare and common structures. A large group of tunes with identical structure, or a tune with unique structure (see examples in Fig. 3), immediately raises new potential research questions, and this is likely the primary use of the application.

In some cases, many tunes have unique structures, but still have long common prefixes. Collapsing levels of the prefix tree structure is a natural way to explore this type of structure and this is possible via the control bar shown in Fig. 4.



Galway

Part: A | Classifications: Double Jig | Tune Count: 365

Figure 4: The control bar allows the user to expand or collapse levels of the tree or to search by structure or tune title (here "Galway"). The "Prepare Graph" button allows the user to load a new dataset in a simple csv format and/or to filter by tune type, e.g. "double jig".

The application was implemented and hosted using Firebase, Node, Typescript, React, and D3. A demo is available online[1].

References

Seán Doherty. Melodic structures in the double jigs of O'Neill's the dance music of Ireland: 1001 gems (1907). *Journal of the Society for Musicology in Ireland*, pages 19–45, 2022.

Francis O'Neill. *The dance music of Ireland: 1001 Gems.* Lyon & Healy, 1907.

[1] Please contact the author for the necessary data files. https://fyproject-98a82.firebaseapp.com/