A Contemporary Realisation of Scambi

Robin Fencott and Simon Harris

Pousseur's original sonic components for Scambi have been converted from lengths of tape to digital audio files. Originally, realising a version of Scambi would have been a physical and interactive experience involving tape cutting and splicing. Now that Scambi's components are in the digital domain, creating a realisation of Scambi is not such an involving task. Sound files can be quickly arranged in a sequencer with little effort or consideration.

It was our intention to develop the realisation process by enhancing the original Scambi scheme into a larger serial structure. This would, in turn, extend its influence into the individual sound components. We were mindful of Pousseur's intentions for Scambi and did not plan to alter or break his rules for continuity. Adhering fully to Pousseur's rules presented us with a system that we could work against.

To improve the quality of the raw files, we applied gating, digital noise reduction and compression, to remove tape hiss and boost the overall level.

Through further processing we wanted to enhance the characteristics (pitch, speed, homogeneity and continuity) that Pousseur defined in the original Scambi system, whilst retaining their original qualities and parameter trajectories. We were strongly against transforming the files beyond recognition, and our serial scheme was designed with this in mind.

We created an eight segment looping Scambi structure. This meant the ending conditions for the final segment matched the starting conditions for the first segment (i.e. A Scambi structure that begins and ends with the values 1100). We decided to use this looping structure as a core template for creating an in-depth serial scheme that would affect the local and global parameters of the piece.

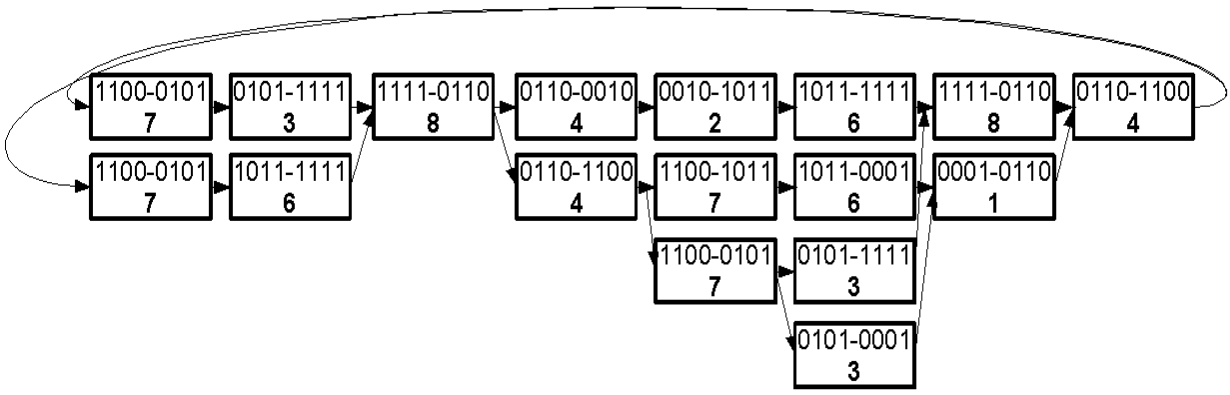

Fig 1: Scambi Structure Diagram

We noted that there were eight starting or ending conditions (e.g.1100), which we numbered, and referred to as the 'Scambi Number' (shown in bold in fig.1).

Start/end condition	0001	0010	0101	0110	1000	1011	1100	1111
Scambi Number	1	2	3	4	5	6	7	8

Fig 2: Table of Scambi Numbers

We created four differing routes through the Scambi Structure, colour coded gold, yellow, green and blue for ease of identification (fig. 3).

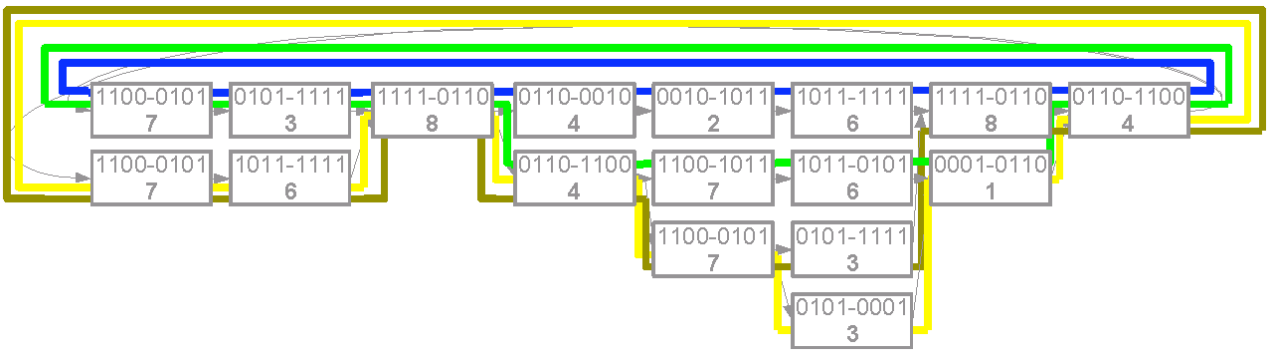

Fig 3: Colour routes

By plotting the start and end Scambi Numbers of each file (i.e. 1100 – 0101 converts to Scambi Numbers 7 – 3) along the designated colour routes, we arrived at four effect parameter graphs (fig.4.).

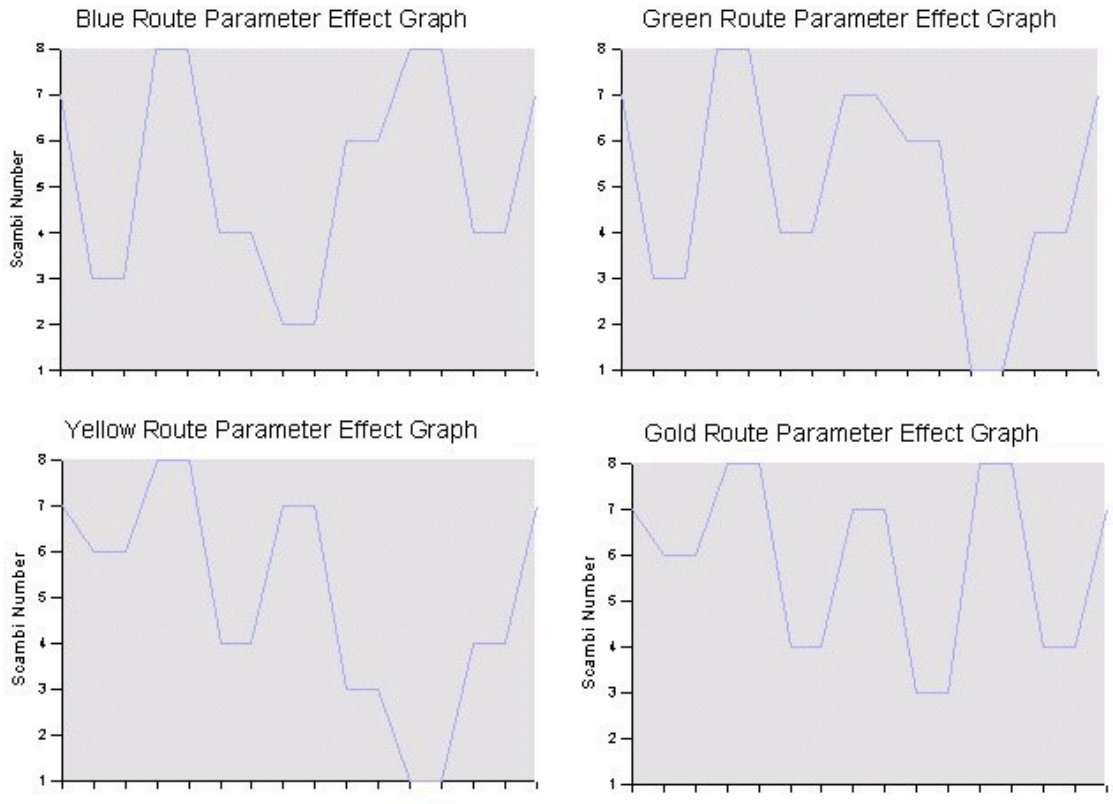

Fig 4: Effect Parameter Graphs, derived from colour routes

We wanted to use the effect parameter graphs to modulate the four parameters that Pousseur originally used to categorise his Scambi sound segments, thus further imposing the Scambi Structure's influence upon our sound files.

We were aware that using the same set of effect parameter graphs on every sound file would result in obvious repetitions or patterns, and to counteract this we added two specific improvements. Firstly, we would use the Scambi Number associated with the file's starting condition, to stipulate the number of effect parameter graph cycles per sound file. For example, a file with a Scambi number 6 would have six effect parameter graph cycles. Secondly, we used the Scambi number to determine the starting point in the effect parameter graph. For example the graph for a file with Scambi Number 2 would start at point two in the graph and would contain 2

graph cycles.

Pousseur used the figures 0 and 1 to define the start and end conditions of his four parameters, for example a file may start with a pitch of 0 (meaning low), and end with a pitch of 1 (meaning high). However, when applying our parameter effect graphs, the files original start and end conditions could be distorted, meaning in this example that the file may end on a low pitch. This would disrupt Pousseur's scheme and render continuity between files impossible.

In order to preserve the file's original audible qualities, as categorised by Pousseur, it was necessary to scale the graphs according to the file's 0 or 1 start and end conditions. This would ensure that any effect changes would fully adhere to the file's original shape.

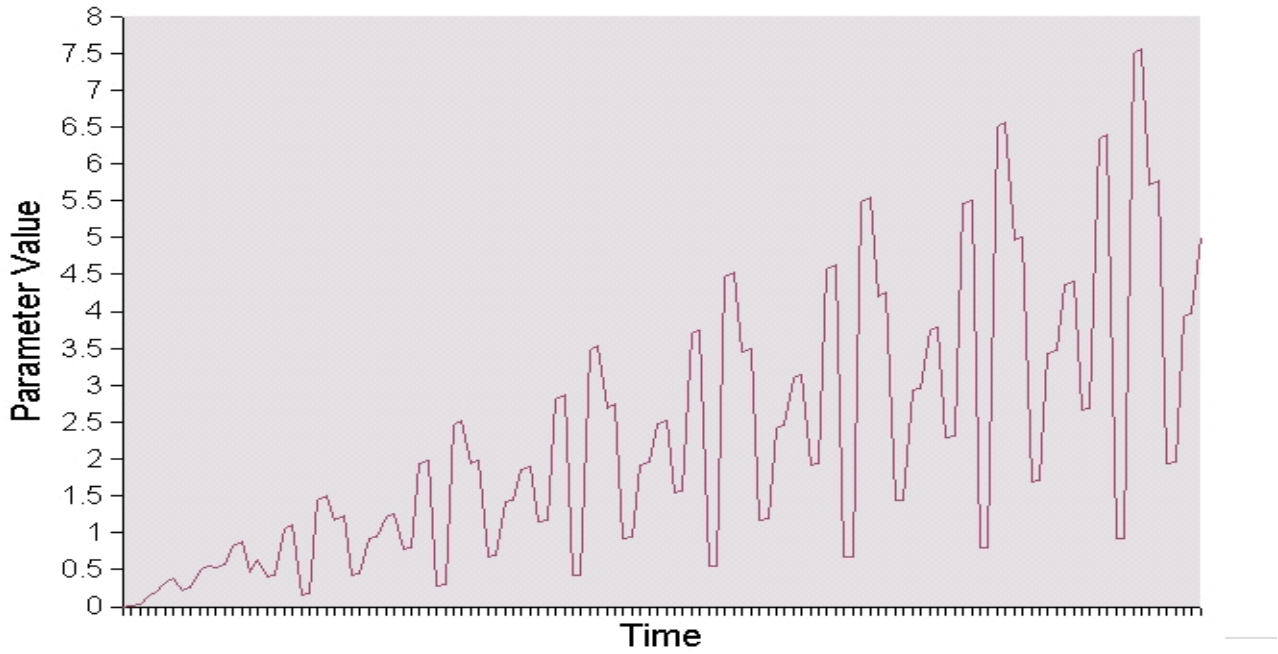

Fig 5: Example showing an Effect Parameter Graph, cycling eight times, and scaled from 0 to 1.

We used four separate processes to apply our parameter enhancements, they are as follows:

Original Parameter	**Parameter Enhancement Effect**
Pitch	Pitch shifting
Speed	Real-time time stretching
Homogeneity	Reverberation
Continuity	Real-time slicing, re-ordering and repetition

It was impossible to accurately implement the parameter modulations manually, so an automated system was constructed to control the effect parameter enhancements. This ensured our serial scheme was realised with a high degree of accuracy.

Whilst applying the effect enhancements we would sometimes create more than one version of a file. It became clear that due to the variable nature of some of our effects that the resulting files were not always identical. We allowed ourselves the artistic licence of picking our preferred version of the processed files.

Once all the files were enhanced and our preferred files chosen, we assembled the new files in a sequencing package in accordance with the original Scambi Structure.

We wanted to be selective about the overall shape of the piece. For example, we felt it was important to have the section featuring four simultaneous files in the middle to create a climax. Due to the looping nature of our Scambi Structure, we were able to choose an appropriate entrance point into the loop that satisfied these conditions.

Once all the files were in place, we manually adjusted the dynamics and applied spatialisation to complement the natural movement of the piece.

One grey area we came across was the difference in sound file lengths. With files being either 42 or 30 seconds long this gave us a dilemma as to how to fill the potential gaps. Obviously with Scambi having been designed to maintain continuity between sounds it was essential that all sound files were joined together. It was decided that files would be split at an appropriate point, such as an obvious end of a phrase or a silence. This ensured that the continuity the system was aiming to create between the sound files would be maintained.