VOCEL
New implementations of the FOF synthesis method

John Michael Clarke*
Peter Manning and Ron Berry
School of Music
University of Durham
Durham
ENGLAND

Anne Purvis
School of Engineering and Applied Science
University of Durham
Durham
ENGLAND

Abstract
A new unit-generator has been written to make FOF synthesis available within Max/MIR. The purpose of this is to help make FOF synthesis more widely available and to explore its potential in a new
alternative environment to that of CHANT. Musical examples will be played to demonstrate certain features of the program. Recently a new version of the subroutine has been written in C (the COSOUND)
Current research is directed towards producing a real-time synthesizer
implementing FOF synthesis by using concurrent processing (transpat-
ency) and VLSI design.

Introduction
FOF ( Fonction d’Onde Formatique) systems were invented by Xavier Rodet
and in the synthesis method used the CHANT program from IRCAM.

*The author is grateful to financial assistance from the Arts Council of Great Britain
and the Swedish University during the course of this project. He is currently continuing this
work as a member of staff in the Music Department, BURDESFIELD POLYTECHNIC,
Queensgate, Huddersfield, ENGLAND.
CHANT has been widely acclaimed for the richness and realism of its sound and FOG synthesis has been an important factor in this success. Until now, however, FOG synthesis has been available only as part of the CHANT package which was designed primarily for vocal imitation. Although many other sounds are possible with CHANT the structure is determined by the vocal model and this imposes certain restrictions on its use. The VOCEL project is intended to help make FOG synthesis available on smaller computers and to present it in an environment where it can be used very flexibly in an "experimental" way.

A FOG Unit-Generator for Music 11

The first stage of the VOCEL project (from which it derives its name: VO(C)e Eleven) was to create a new FOG unit-generator for Music 11. This takes the form of an extended subroutine (some 200 lines of code) written in Music 11 Assembler. Music 11 has been described as having a "patch/parameter" structure in contrast to the "synthesis-by-rule" approach of CHANT. One result of this is that whereas in CHANT the FOGs are part of a fixed configuration (the user cannot change the input to the configuration or even add to it at certain points by means of "user-subroutines" but cannot change in shape) in VOCEL FOG unit-generators are modular just like any other Music 11 unit-generator and the user can design almost any patch he wishes. While gaining in flexibility in this way VOCEL loses in ease of use where vocal imitation is concerned; the many automatic calculations of CHANT designed to reproduce various vocal characteristics are not pre-programmed in VOCEL and the user must specifically build them into the patch. However, it is possible to build such vocal-imitation patches and the usual results are nearly comparable to those of CHANT. Appendix 1 to this paper is a Music 11 orchestra file demonstrating such a patch, and the following musical examples demonstrate some of the voice-like sounds possible with VOCEL.

[Example notes]

A simplified schematic representation of the fixed configuration of FOG’s in CHANT is shown in Ex.1. Each of the FOG generators creates a different formant region and their outputs are summed to give the overall spectrum. In VOCEL Ex.1 represents just one of the many possible configurations. A small but potentially interesting variant perceived by VOCEL is in each
FOF to have an independent fundamental frequency [Ex.2a] or more subtly to give each FOF a common fundamental but with the possibility of diverging [Ex.2b]. In this way a single voice sound can split into independent streams or vice versa.

[Musical examples 2a, 2b]

Of course completely new configurations can also be explored in VOCELI for example, the output of one FOF generator might even be used to modulate one of the inputs of another FOF generator [Ex.3].

FOF synthesis theory—a summary

Before describing in more detail certain original features of VOCELI's FOF generators, it might be helpful to outline the theory behind FOF synthesis. This is however only a summary as a detailed account is already published [Rodet 1984; Rodet, Potard and Barrière 1984].

Each FOF synthesis unit guides just one formant region. The unit generates a series of excitations, an excitation comprising a sine wave at the centre frequency of the formant band [Ex.4a] amplitude modulated by a "local envelope" [Ex. 4b] (so-called because its duration is normally of the order of 20 milliseconds and therefore considerably shorter than the larger "global envelope" of a note). It is this "local envelope" which determines the spectral envelope around the formant centre frequency. Its rate of decay controls the bandwidth of the spectrum at -6dB while the rise rate determines the wider skirt-width at -40dB [Ex.5]. FOF synthesis thus permits very detailed shaping of the spectral envelope of formant regions.

The fundamental frequency of such a FOF unit is the raw at which new excitations are produced. With all yet the lowest fundamentals new excitations will overlap with earlier excitations still completing their course. Ten or more overlapping excitations may in fact be required from each FOF generator in the production of a high vocal sound and hundreds of such overlapping excitations can be used to produce integrating new timbres as X. Rodet and J.-B. Barrière have proved. In CHANT such overlapping was not a problem as the whole program was specifically designed for this purpose. Creating the overlapping structure within an existing program (Music 11) was more difficult but a satisfactory solution was found using nested do-loops with a number of alternative branches. However, because memory has to be allocated to the unit-generator in advance, it is scenery in Music
11 to specify the maximum number of overlapping excitations required as a parameter at initialization.

Appendix 2 contains examples of time-series generated by the VOCEL 70F unit-generator, each followed by its FFT.

Unit-Generator Specification

The VOCEL unit-generator contains some original features not present in the original CHANT program. Ex. 5 gives its specification in the style of the Music 11 manual.

"x" is the result or output of the unit-generator.

The first parameter is the "op-code": the name by which the unit-generator subroutine is called in the program.

The next parameter is the "op-code": the name by which the unit-generator subroutine is called in the program.

The first two input parameters control the amplitude and fundamental frequency of the generator respectively.

Input parameters 3 and 4 both control the formant frequency. In CHANT the formant frequency will only have effect when a new excitation occurs. This feature can often result in rich textures with several overlapping excitations each having fractionally different (but fixed) formant frequencies. In VOCEL the option is available using input parameter 3 "shift". As an alternative, a continuously variable formant frequency, was also needed in VOCEL to permit, for example, frequency modulation of the formant frequency. Input parameter 4 "dither" provides this alternative. In fact the two parts are summed to give the actual formant frequency as a combination of the two methods is possible.

Input parameter 5 "icon" permits "multiple-activation" of the FOF signal. Octivation is the gradual fading out of alternate excitations eventually halving the rate at which new excitations are produced and so halving the fundamental frequency. Ex. 7 shows this process schematically. The musical result is quite different from a glissando; the transformation is more like a change in timbre than in pitch. Single-activation is available in some of the standard CHANT user-subroutines but while working with CHANT x EMS in Stockholm Clarke wrote his own user-subroutine to extend this principle to a range of eight octaves (over alternate excitations have been removed from the original signal, alternate excitations are again faded out from those that remain...and so on). Later's ten-octave version was incorporated in VOCEL's unit-generator. Using this technique it is possible to move from a high soprano sound to the slow climbing of the glissato as can be
heard in this example:

[musical example 3]

By varying other parameters while this process occurs it is possible to end up not with clicking but instead with something more bell-like.

[musical example 4]

If the same technique is applied to a texture rather than a voice sound (the texture being produced by random variation of certain other FOF parameters) the result can be a quite intriguing textural evolution. An example of this sort of process will be played later in the paper.

The next four input parameters control the spectral envelope of the formant region. They correspond directly to parameters in CHANT (tess, band, attack, and atten respectively).

Of the remaining input parameters two deserve comment as they vary from the practice of CHANT. "fina" and "flash" identify the user-defined lookup table used to generate the FOF signal. "fina" is normally a stored sine-wave table used to produce the formant sine wave for each excitation [Ex. 4a]. "flash" is the wave table used to shape the attack portion of the local envelope [Ex.4b] and also its final decay. Both parameters will normally reference the same sine-wave lookup table and in CHANT no alternative is available. In VOCEL it is possible to change either of these wave forms for special effect or to create more complex timbres.

[musical example 5]

FOF Synthesis and Granular Synthesis

Some of the more "experimental" uses of FOF synthesis begins to produce timbres and textures similar to those normally associated with granular synthesis. Each individual FOF excitation is in fact similar to a granule in granular synthesis (C. Roads, 1978) and CHANT might be considered a specialized form of granular synthesis. One of the purposes of VOCEL with its flexible configuration is to forge clear links between FOF synthesis and granular synthesis and to enable smooth transitions between voice-like timbres and granular textures. Multiple excitation is particularly useful in this respect as it can lead to a blurring of the normal distinctions between pitch, timbre and texture. The next musical example gives just a hint of what might be possible in this respect.
Such radical transformations depend on being able to “get inside” the structure of a sound in a way most synthesis or processing methods do not permit. It is one of the outstanding features of the FOF synthesis method.

Assessment of the Project and Future Plans

The Macro 11 subroutine for Music 11 has proved successful in terms of quality and flexibility of use. A version of this subroutine in C has now been produced. The main problem with both subroutines has been the slow speed of calculation. FOF synthesis is time consuming and running it within a general purpose package increases the problem (the speed however varies greatly with the simplicity of the configuration chosen, the number of overlaps required and, of course, the hardware available). The problem of speed has stimulated several major new developments in the project which are currently in progress. One of these is collaboration with M. Rodet to design a special VLSI chip for use in FOF synthesis at Durham University. Another project led by Purvis and Manning is exploring the use of transputer and concurrent processing to achieve higher speeds. Tests have been conducted using transputer arrays to produce oscillators and elementary FOF generators in real-time. The eventual objective of this work is to produce a real-time programmable synthesizer in which transputers control VLSI chips. The synthesizer is intended to be flexible and, following the example of Music 11, allow the user to create “patches” out of separate modules. The design of transputers and of the OCCAM programming language seems ideal for the control of such a synthesizer. However, thorough speed tests conducted by Berry have indicated that the synthesizer itself is better carried out by dedicated VLSI chips.

Another development stemming from this project Clark has just completed at the time of the ICMC five months work at IRCAM helping to develop FOF and granular synthesis on the Sun-Mercury workstation.

Acknowledgements

The authors wish to thank X. Rodet and Y. Potard of the original CHANT team at IRCAM for their help and encouragement in this project and also B. Vercree of MIT, author of Music 11 and SOUND, for his support.
References


Sound Examples

All the sound examples in this paper (except 1a) come from the compositions "Milanskeg" by J.M. Clarke or from the sketches for this work. The work was composed using the CHANT program, with the addition of composer's own subroutines, and Music 11, with the addition of VOCSEL's FOF unit-generator.
Diagram 1: Illustration of CHART (amplitude).

Diagram 2: Illustration of CHART (frequency).

Diagram 3: Spectral analysis of a signal region.

Diagram 4: Illustration of VCA1, which shows phase-locking from a common fundamental frequency.

Diagram 5: Spectrum representation of an ideal process.

Diagram 6: Spectral representation of an actual process.

Diagram 7: Spectrum representation of an actual process.
Ex. 1 Structure of CHANT (simplified)
Ex 2b: VOCEL patch allowing divergence from a common fundamental frequency

Ex. 3 Output of one FOF generator modulates an input of another
Ex. 4 Formant Wave Function

Ex. 5 Spectral envelope of a formant region
Ex. 7 Schematic representation of octavation process
APPENDIX I

IA BASIC VOICE INITIATION PATCH
instr 1
    n1 = 1/(d2+d3+d4+d5+d6+d7); amplitude factor for all formants
    t11 = n1; icorrection of 1st formant freq
    if t11 < p5 goto cor1
    goto next
    cor1: t12 = p5
    next: t12 = p7
    if t12 > 300 goto follow
    if p5 < 200 goto follow
    t12 = (p5-200)*2/3/(t12-1300)/7000
    follow: if t12 >= (2p5p3)+30 goto env
    t12 = (p5p2)+30
    igLOBAL ENVELOPE
    env: a1 linseg 0 ,.01,p4 ,p5,.02 ,p4,.7 ,.01 ,0
    iVIBRATO AND RANDOM VARIATIONS OF FUNDAMENTAL
    k20 randi .012 ,1/-.91
    k21 randi .015 ,1/1.7
    k22 randi .015 ,1/1.015
    k23 randi .015 ,1/-93
    k24 = 0.02*(k20+k21)
    k25 = 0.01*(k22+k23)
    k2 oscil p5 ,5.1+k25 ,1 iVibrato
    k2 = k2*(.02+k24)
    a2 *randi .01 ,1/-.05
    a3 randi .01 ,1/1.111
    a4 randi .01 ,1/1.2186
    a5 = a2+a3+a4+a5
    adjusted fundamental frequency
    ifOF UNIT-GENERATORS
    i amp fund fua,b oct twc band deband atten olpm fua,b dur phs
    a'f of p12 ,a5 ,t11 ,0 ,1 ,.003 ,.77 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    a10 of p13 ,a5 ,t12 ,0 ,1 ,.003 ,.88 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    a11 of p14 ,a5 ,p6 ,0 ,1 ,.003 ,.122 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    a12 of p15 ,a5 ,p9 ,0 ,1 ,.003 ,.127 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    a13 of p16 ,a5 ,p10 ,0 ,1 ,.003 ,.137 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    a14 of p17 ,a5 ,p11 ,0 ,1 ,.003 ,.70 ,.01 ,.007 ,p18 ,1 ,1.2 ,0
    isUMMATION OF FORMANTS AND OUTPUT
    a15 = a2+a3+a4+a12+a13+a14
    outs a13,a13,x1,a5,x1
endin

ip4 : amplitude (overall)
ip5 : fundamental freq (Hz)
ip4 to p11 : formant frequencies (and "freq")
ip12 to pi7 : relative amplitudes of above
ip18 : maximum number of overlaps required

371