



Uludağ Üniversitesi Eğitim Fakültesi Dergisi

<http://kutuphane.uludag.edu.tr/Univder/uufader.htm>

Çoksesli Bir Müzik Parçasını Seslendirmek İçin Solisti Kendi Sesiyle Gerçek Zamanda Çokseslendiren Bir Algoritmik Kompozisyon Yöntemi ve Örnek Bir Uygulama

Uzay Bora

Dokuz Eylül Üniversitesi Devlet Konservatuvarı, uzaybora@gmail.com

ÖZET

Bu çalışmada, çoksesli bir parçanın icrasını, bir solistin sesini bilgisayar yardımıyla gerçek zamanda kendisiyle çokseslendirerek organize etmek için tasarlanan bir yöntem sunulmuş ve bu yöntemin Max programlama ortamında uygulanmasıyla oluşturulan örnek bir algoritmik kompozisyon tanıtılmıştır. Çalışma, bilgisayar teknolojisi desteğiyle müzik performansı için yeni olanaklar sunabilecek bir tasarım örneği oluşturmayı amaçlamaktadır. Yöntemin açıklanmasının ardından, programı tamamlayıcı beste ve uygulandığı özetle tanıtılmış, sonuçlar verilmiş ve gelecek çalışmalar için önerilere yer verilmiştir.

Anahtar Sözcükler: Bilgisayarlı Müzikoloji, Bilgisayar Destekli Müzik Performansı, Algoritmik Kompozisyon.

An Algorithmic Composition Method for Performing a Polyphonic Musical Piece by Generating Polyphony from a Soloist's Own Voice in Real-Time and a Sample Application

ABSTRACT

In this study, a method designed for organizing a performance of a polyphonic piece by rendering a soloist's voice polyphonic with itself in real-time with the aid of a computer is presented and a sample algorithmic composition produced by the implementation of the method in Max programming environment is introduced. The purpose of the study is to produce a sample design that may offer novel possibilities to music performance with the aid of computer technology. Following the explanation of the method, the composition complementing the program and its application are briefly described, and the results and suggestions for future studies are given.

Key Words: Computational Musicology, Computer Aided Music Performance, Algorithmic Composition.

GİRİŞ

Bilgisayar teknolojisinin müzik uygulamalarında olanaklı kıldığı işlemler arasında, seslerin süresi değiştirilmeden perdesinin, perdesi değiştirilmeden de süresinin değiştirilebilmesi, bu alanda *phase vocoder* ve granüler sentez yöntemlerinin ortaya çıkışından bu yana bilinen bir konudur. Bu çalışmada, süreyi değiştirmeden perdeyi değiştirme işlemiyle, gerçek zamanlı ('anında'; *real-time*) olarak bir kaynak sestemden çeşitli perdelere aktarılan (transpoze edilen) sesler elde ederek ve bunları performans sırasında kaydetme, yeniden çalma, aktarımlı çalma, üstüste toplama gibi işlemlerle organize ederek çoksesli bir parçanın seslendirilişini gerçekleştirmek için bir tasarım sunulmaktadır. "Çoksesli" nitelemesi burada dar anlamıyla kullanılmıştır; parçanın aynı anda birden fazla farklı ezgi partisi içeren (polifonik) dokusu olduğunu belirtmektedir. Çalışmada, çoksesli bir parçayı tek bir sopranonun bilgisayar desteği ile gerçek zamanda kendi sesiyle çokseslendirerek yorumlayabilmesi düşüncesi bağlamında, teknoloji yardımıyla müzik performansında yeni olanaklar ve deneyimler sunma potansiyeli olan bir tasarım örneğinin ortaya konması amaçlanmaktadır.

Kaynak sesin, bir çalgı sesi yerine vokal olarak seçilmesinin amacı, solistin söylediği hecelerin üretilen eşlik¹ partilerinde de değiştiğini, böylece gerçek zamanlı işleyişi (parçanın önceden örneklenmiş seslerle çalınmadığını) duyurmaktır.

Örnekleyici (*sampler*) özelliği bulunan bir *synthesizer* kullanıldığında, genellikle “orta do” olarak kaydedilen ya da bu perdeye akortlanan bir ses örneği, istenen perdelere aktarılır. Ancak bu aktarım, özellikle eski uygulamalarda, örnek üzerinde yalnızca bir *resampling* (yeniden –başka bir hızda– örnekleme) ya da dijital ortamda matematiksel olarak buna eşdeğer olan “örneği başka bir hızda çalma” işlemi ile yapıldığından, devir düşmesi ya da hızlanması olgusunda olduğu gibi, eğer aktarım kalın bölgeye (aşağıya) doğruysa ses yavaşlar, inceldikçe ise hızlanır (bkz. Duncan and Rossum, 1988; Rossum, 1989).

Aktarımları sesin süresini koruyarak yapabilmek için kullanılan teknikler, zaman bölgesinde çalışanlar ve frekans bölgesinde çalışanlar (ör. *phase vocoder*) olmak üzere ikiye ayrılmaktadır. Zaman bölgesi (*time domain*) tekniklerinin, sesin bozulmaması için perde saptamaya çalışmaları gerekmektedir. Bu süreç, girdinin tek sesli (monofonik) olması zorunluluğunu doğurmaktadır. Perdelerin saptanması, başarısı işlenen kaynağın özelliklerine sıkı biçimde bağlı ve sürekli doğru çalışma garantisi olmayan bir süreçtir. Ayrıca zaman bölgesi teknikleri, $\pm 15\%$ ’lik bir frekans ölçekleme farkını aşan perde ötelemeleri için yetersiz kalmaktadır (Dorran et al., 2006, s.16). Frekans bölgesi (*frequency domain*) tekniklerinin ise, kaliteli sonuç üretmelerinin karşılığında hesaplama yükünü arttırdıkları, gerçek zamanlı uygulamalar için gecikmeye yolaçtıkları belirtilmiştir (Bernsee, 1999; Haghparast et al., 2007). Bununla birlikte, bu çalışmada hem uzak aralıklara aktarım yapıldığı, hem de parçanın bazı yerlerinde çoksesli kısımların, yani birkaç partinin birlikte aktarılması tercih edildiği için, frekans bölgesinde çalışan bir tekniğin seçilmesi uygun olmuştur. Aktarımın göreceli yapılması sayesinde, uygulanan bir parçanın herhangi bir tona aktarılma işleminin önceden program ayarları değiştirilerek ya da perde izleyerek yapılmasının gerekmemesi, her solistin kendisine uygun bir tonda seslendirebilmesi amaçlanmıştır. Gecikmelerin etkisini azaltmak için başvuru olan yollar aşağıdaki altbölümlerde açıklanmıştır.

Vokal ses ve gitar sesi gibi kaynakları çokseslendirmeye yönelik bazı popüler ticari yazılım ve donanımların varlığı da burada anılabilir.

¹ Eşlik terimi, çoksesli dokudaki ayrı ezgi partileri için normal olarak kullanılırsa da, bu yazı boyunca anlatımı çok yalınlaştırdığından, üretilen diğer partileri belirtmek için kısaca eşlik denmiştir.

Genellikle giriş sesine dikey armoni uygulayan bu gibi araçların çoğunda akor değişimleri ayarlanamamakta, bir kısmında sabit ayarlar içinden “armonizasyon tipi” ya da sabit aralık seçimi yapılabilmektedir. Bazıları yazılımdaki vokal sistem modelleri doğrultusunda ‘başka’ (kaynak sesi aktararak elde edilmeyen) vokaller sentezlemektedir. Gerçek zamanda çalışanların tümü zaman bölgesi tekniklerini kullandığından, ilgili kısıtlama ve perde izleme sorunlarını içermektedir (Ticari marka belirtmemek için referanslar verilmemiştir).

İzleyen bölümlerde, seçilen tekniğin kullanıldığı bir tasarımın Max programlama ortamında uygulanışı, örnek bir bestenin tanıtımı ve sonuçlar verilmiştir.

YÖNTEM

Bir “kaynak” sestem yararlanarak iki ezgisel parti oluşturmak istendiğinde, referans olarak kullanılacak bu kaynak ses eğer –örneğin– bir *sampler*da kayıtlı bir ses ise, bu sabit referansa göre aktarımların uzaklıklarının Şekil 1a’daki gibi hesaplanacağı bellidir.

Figure 1a shows a musical score with two staves. The top staff is in treble clef and the bottom staff is in bass clef. The key signature has two sharps (F# and C#). The time signature is 4/4. The score consists of two measures. Above the first measure, there are two upward-pointing arrows with the numbers +16 +15 +18 and +12 +13 +9 +11. Below the first measure, there is a downward-pointing arrow with the numbers +3 +1 -2 -11 -7 -9 -8 -9 -15 -13 -11 -11. The text "(ORTA DO'YA GÖRE)" is written below the first measure. The second measure has no annotations.

a

Figure 1b shows a musical score with two staves. The top staff is in treble clef and the bottom staff is in bass clef. The key signature has two sharps (F# and C#). The time signature is 4/4. The score consists of two measures. Above the first measure, there are two upward-pointing arrows with the numbers +9 +8 +15 and +9 +8 +3 +6 +8. Below the first measure, there is a downward-pointing arrow with the numbers -3 -6 -9 -14 -8 -10 -9 -9 -18 -18 -17 -16 -14. The second measure has no annotations.

b

Şekil 1a: Sabit Bir Referansa Göre Aktarımlar
b: Değişken Bir Referansa Göre Aktarımlar

Burada aralıklar “yarım ses” cinsinden belirtilmiştir. Buna göre, örneğin tam beşli $\equiv 7$, unison $\equiv 0$, küçük yedili $\equiv 10$ gibi.

Referans olarak kullanılacak kaynak, bu iki (karşı)ezgiyle birlikte varolacak bir “ezgi” olduğunda ise, referans sürekli olarak değişmektedir. Bu durumda aralıkların hesaplanması Şekil 1b’deki gibi olur. Bu çalışmada, mikrofondan alınan vokal ezgi, diğer ezgileri de oluşturmada kullanıldığı için, aralıkların hesaplanış yolu Şekil 1b’deki gibidir.

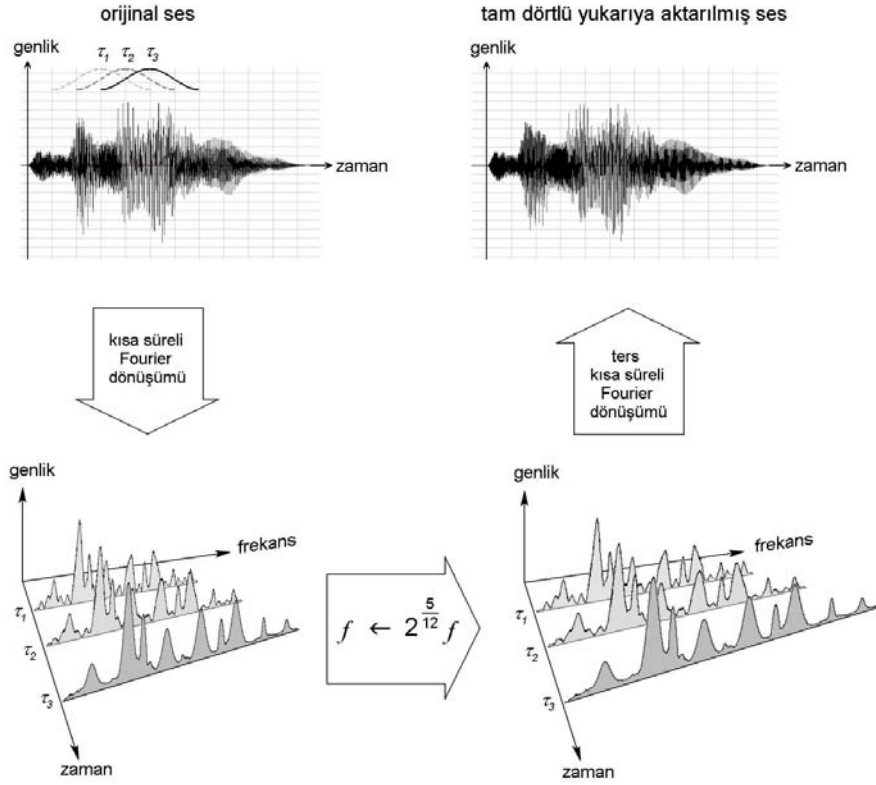
Kuramsal Altyapı

Özellikle sinyal işlemecilerin yakından tanıdığı bir olgu, zaman ve frekans bölgelerindeki çözünürlüklerin ikisinin birden çok arttırılamayacağı, birinin daha ayrıntılı incelenmesinin diğerinde ayrıntı kaybına neden olduğudur.

Rioul ve Vetterli (1991, s.17), en küçük zaman değişimi ve en küçük frekans değişiminin çarpımlarının $1/(4\pi)$ ’ye eşit ya da daha büyük olması gerektiğini göstererek bunun belirsizlik ilkesi, ya da Heisenberg eşitsizliğinin bir görünümü olduğunu belirtir.

Yukarıda belirtilen nedenle, sesin kısa süreli spektrumunu hesaplamak için zaman boyunca çözümlenen kısmın (bunun için kayan bir ‘analiz penceresi’ kullanılır) frekans netliğini yitirmeyecek bir uzunlukta seçilmesi gereklidir. Ses verilerinin, ancak sözkonusu uzunlukta blokların geliş (belleğe alınma) süreleri beklenerek işleme konulabilmesi, gecikmenin doğal nedenidir.

Bu çalışmada kullanılmak üzere seçilen, “frekans bölgesinde perde öteleme (*frequency domain pitch shifting*)” tekniği özetle Şekil 2’deki gibi şematize edilebilir. Şekilde örnek olarak, bir sesi tam dörtlü (5 yarım ses) yukarıya aktaran dönüşümler dizisi simgelenmiştir. (Şekil 2’de dönüşümler karmaşık faz (evre) gösterilmeden, mutlak genlik değerleriyle belirtilmiştir. Sesin bozulmasını önlemek için faz bilgisinin nasıl kullanıldığı konusu bu makalenin kapsamı dışındadır; ilgili bir yazı için bkz. Bernsee, 1999.)



Şekil 2: Frekans Bölgesinde Perde Öteleme

Çalışmadaki Tasarım ve Program Bileşenleri

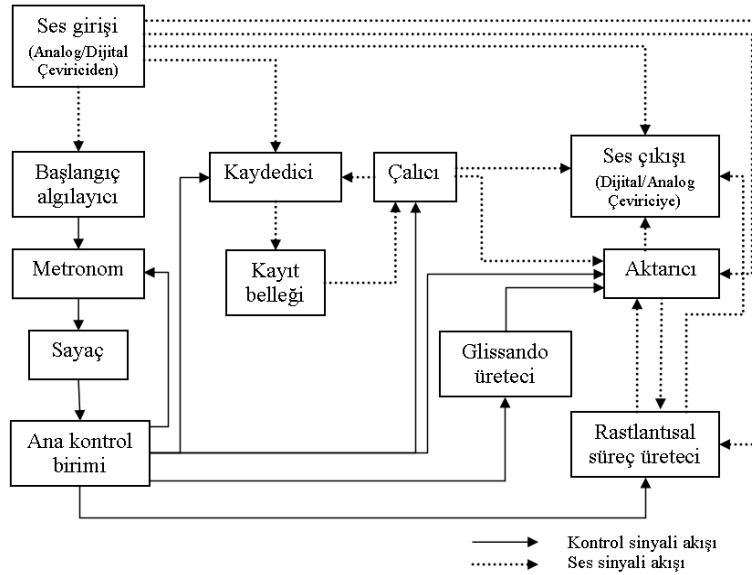
Düzenegi oluşturan malzemeler mikrofon, 2,7 GHz işlemcili güncel bir sistem içeren bilgisayar ve hoparlörlerdir.

Parçanın seslendirilebilmesi için organize edilen algoritmik kompozisyon, Max programlama ortamında geliştirilen yazılımla gerçekleştirilmiştir. (Daha önceki bir algoritmik kompozisyon tasarımı için bkz. Bora, 2010; algoritmik kompozisyon kavramı üzerine geniş bir tanıtım için bkz. Bora, 2003; Max² için bkz. Puckette, 2002.)

² Max, belli işlevleri olan çeşitli nesne tipleri, bunların giriş/çıkış bağıntıları, nesnelerin bağlanabilirlik seçenekleri gibi öge ve kuralları içeren, ağırlıklı olarak grafiksel bir programlama ortamıdır. Programcı, hem Max'in kütüphanesindeki nesnelere, hem de program yazarak oluşturduğu kendi nesnelere kullanabilir.

Kaynak sesi aktarmak için frekans bölgesinde perde öteleme tekniğini kullanan bir Max nesnesi olan *gizmo~* ile çalışılmıştır. (Vokal girdiyi çokseslendirmek için Jehan'ın (2007) tasarladığı bir Max nesnesi bulunmaktadır, ancak tek ses kısıtlılığı, perde izleme ve üretilecek partilerle ilgili bilgiyi MIDI kontrolü ile verme zorunlulukları nedeniyle buradaki tasarıma uymamaktadır).

Çalışmadaki tasarım, en genel hatlarıyla Şekil 3'teki blok diyagram ile simgelenebilir. Programın işleyişi şöyle özetlenebilir: Mikrofondan alınan sesin 300 ms boyunca belli bir şiddetin üzerinde olması, başlangıç algılayıcı algoritmanın, parçanın seslendirilmeye başladığına karar vererek metronom ve sayaç nesnelerini çalıştırmasını sağlar. Sayacın oluşturduğu zaman bilgisinin iletildiği ana kontrol birimi, parçadaki zamanlamalarına göre her bir olayı işleme koyar. Metronom atımları, parçadaki en kısa nota değerine göre ayarlanır. Bu çalışma için bestelenen örnek parçanın çoğu kısmında bu değer sekizlik, birkaç yerde ise üçleme sekizliktir. Tüm ayarlar gibi bu hızlar arasında zamanı geldikçe geçiş yapılmasını da ana kontrol birimi yerine getirir. Bu birimin kumanda ettiği diğer öğeler arasında kaydediciler, çalıcılar, ses düzeyi ayarları, aktarımları yapan altmodüller ve akış kontrolünü sağlamaya yönelik çok sayıda nesne sayılabilir. Örnek besteye özgü olarak, *glissando* ve rastlantısal özelliklerde eko (yankı) üreten altmodüller de programa eklenmiştir.



Şekil 3: Tasarımın Blok Diyagramı

Programın ana modülü Şekil 4'te gösterilmiştir. Ana kontrol birimi işlevindeki nesne (Şekil 4'te "js zamanizle1.js"), javascript ile programlanmıştır. Aktarımı (frekans ölçekleme işlemini) yapan *gizmo~nesneleri*, frekans bölgesine geçmeye ve zaman bölgesine geri dönmeye yarayan dönüştürücü nesne çiftini içeren altmodüllerde çalışabilmektedir. Şekil 4'te başlangıç algılayıcı kısmın yanı sıra, açılmış pencereler içinde ana kontrol biriminin ve aktarıcı birimlerden bir tanesinin içeriği görülmektedir.

Rastlantısal özellikte ekolar üretmek için değerleri rastgele seçilen parametreler, eko zamanlamaları, aktarımları ve şiddetlerine ilişkindir. Parçada ekoların kullanıldığı pasajda, birlikte üretilen 3 ayrı eko için 3 farklı süre aralığı içinden seçilen ilk gecikme süreleri, ivme seçenekleri (sabit hız (ivme=0); hızlanma; yavaşlama), aktarım seçenekleri (aktarım yok; 1 oktav yukarı; 2 oktav yukarı) ve 5 tip dinamik kontur seçeneği (hafifleyen; şiddetlenen; sabit; önce şiddetlenip sonra hafifleyen; önce hafifleyip sonra şiddetlenen) program tarafından rastgele süreçlerle seçilmektedir. Bunun için tasarlanan altmodül Şekil 5'te verilmiştir.

Ekoların üretimi ile ilgili kritik bir uygulama olarak, ekoların kullanıldığı pasaj armonik bölgelere ayrılarak her armoni değişiminde ekolar yeniden başlatılmış, böylece armonik yapının bozulmaması amaçlanmıştır.

Gecikmenin etkisini azaltmak için başvuru iki yol: (1) tempoyu çok hızlı almamak ve (2) performansın başladığı saptandıktan sonraki ilk vuruşta, eşliklerin çalınma zamanlamalarının tümünü analiz penceresi uzunluğunun yolaçtığı sabit gecikme kadar öne almaktır. Birinci önlemin nedeni, gecikme süresi sabit olduğu için, parça ne kadar yavaş olursa gecikmenin bir nota süresinin o kadar küçük bir yüzdesine indirgenecek ve müziğin algılanışını o kadar az bozacak olmasıdır. İkinci önlem ise, solistin parçanın başından itibaren tempoyu izleyerek eşzamanlı kalabilmesi içindir. Bu şekilde erkene alınan eşlik partisi zamanlamaları, kayıtların çaldırıldığı pasajlarda ise 'yerine' alınmıştır.

Tasarımdaki etkileşim boyutu, soprano ve bilgisayar arasında aynı anda karşılıklı olarak değil, sırayla gerçekleşmektedir: Parçanın başlangıcında solistin mikrofonun alabileceği bir konumda performansa başlaması ile program performansın başladığını saptayıp gerekli modülleri devreye sokmakta, daha sonra ise solist üretilen seslerin perde ve zamanlamalarını dikkatle izleyip performansını buna uygun olarak sürdürmektedir. (Etkileşimin aynı anda karşılıklı olarak da gerçekleştiği bir algoritmik kompozisyonun tanıtımı için bkz. Bora, 2010.)

Örnek Bir Uygulama: “İnanılmaz Soprano”

Bir uygulama örneği olarak, bir sopranonun yukarıda açıklanan yöntemle dört sesli söylemesi amaçlanan bir füg bestelenmiştir: “İnanılmaz Soprano”. Yazar tarafından bestelenen füg, “konuların baştan itibaren birarada bulunduğu çokkonulu fügler” kategorisine giren bir “Çift Füg (Double Fugue)”dür. Toplam 54 ölçü olan parçada, 4 ölçü uzunluğundaki birinci ve ikinci konuların ilk duyuluşundan sonra 5. ölçüde iki konuyla birlikte karşıkonu (*counter-subject*) gelir (bkz. Şekil 6). Solistin parçayı kendi isteğine bağlı olarak *scat* ve opera tekniklerini dönüşümlü şekilde kullanıp doğaçladığı hecelerle seslendirmesi planlanmıştır.

The image displays a musical score for a piece titled "İnanılmaz Soprano". It consists of three systems of music. The first system is for the Soprano part, starting with a tempo marking of ♩ = 96 and a dynamic marking of *mf* *scat / opera*. The second system is for the Piano part, starting with a dynamic marking of *piu forte*. The third system is a continuation of the Piano part. The score is written in treble clef with a key signature of one flat (B-flat) and a time signature of common time (C). The Soprano part features a melodic line with various intervals and rests, while the Piano part provides a harmonic accompaniment with chords and moving lines. The score is labeled with 'a' and 'b' at the bottom of the second and third systems, respectively.

Şekil 6a: 1-4. Ölçüler: Birinci Seste Birinci Konu, İkinci Seste İkinci Konu;
5-8. Ölçüler: Birinci Seste Karşıkonu
b: 17-20: Karşıkonunun Başaşağı Gelmesi (*Inverse*), Son Ses Dışında

Füg dört sesli olduğu için, birinci sergi, iki konunun herbir partide birer kez yer almasıyla, üçüncü sesin ve dördüncü sesin girişinden önce arada episodlar da bulunmadığı için 16 ölçüde tamamlanır. Kaynak sesin partisindeki hareketlerin, bu perde değişimi anlarında üretilen diğer partilerin netliğinde anlık pürüzler oluşturmaması için hareketli (bir anlamda

‘düzensiz³’) partilerin değil durağan partilerin çoğunlukla soliste verilmesi gerekmektedir. Böylece, uygulamanın gereği olarak, birinci konu parçanın genelinde solistin partisine verilmiştir. Birbirine eşdeğer seslerin (sözgelimi birinci ve ikinci soprano) parti aşımı yapabilmeleri kuralının (bkz. Rimsky-Korsakov, 1964, s.140) yanı sıra, seslerin aslında farklı soprano bile olmadığı, aynı kişi olduğu gözönüne alınarak, konuların geleneksel anlamda beklenen partiler dışındaki partilerde gelmesinin birinci serginin sunulduğu açısından sakınca oluşturmadığı kabul edilmiştir.

Parçanın temposu hızlı olmamakla birlikte, herbir sekizlikte oluşan armoni değişimleri, kalıcı bir tonal merkezin olmadığı, tonalitenin sürekli değişim halinde olduğu bir armonik yapı oluşturmaktadır. Episodlarda armoni göreceli olarak biraz daha durağan tasarlanmıştır (ör. bkz. Şekil 7).

30. ölçü: E⁰ (sol melodik minörün altıncı modu) 33. ölçü: AΔ(#5) (fa# melodik minörün üçüncü modu)
31. ölçü 3. dörtlük: G#Δ (sol# melodik minörün birinci modu) 34. ölçü: $\frac{A}{E\#tr}$ (polikord: la majör / mi# majör)
32. ölçü: C#⁰ (mi melodik minörün altıncı modu)

Şekil 7: 30-34. Ölçülerdeki Armonik Yapı

Ses kalitesinde bozulmalar olmaması için kaynak sesin, üretilenlerden daha durağan olması gerektiği yukarıda belirtilmişti. Ancak solistin partisine de hareket getirme amacıyla, birinci konudan çok daha hareketli olan ve toplam 8 kez çeşitli aktarım ve dönüşümlerle gelen karışık konu, 3 kez solist partisine alınmıştır. Bu kısımlarda eşlikler solistten üretilmeyip, önceki ölçülerin kaydının aktarılacak şekilde çalınması ile elde edilmiştir. Bu kısımlardan ilki Şekil 6a’da görülmektedir. Son dörtlük dışında, 2. ve 3. seslerin 5-8. ölçüler arasındaki kısmı, 1. ve 2. seslerin 1-4. ölçüler arasındaki kısmının tam dörtlü aşağıya aktarımıdır.

³ Fiziksel anlamda, perdesi diğer bir sesinkine göre daha hızlı değişen bir ses daha ‘düzensiz’ olarak nitelenebilir.

Amaçlanan performansa ne kadar yaklaşıldığını ayrıntılı olarak inceleyebilmek için, programa parçanın her icra edilmesinde bilgisayarın ses çıkışına gönderilen toplam veriyi kaydeden bir birim eklenmiştir. Rastlantısal süreçlerin eklendiği pasajın her performansta nasıl değiştiğini de karşılaştırarak izleyebilmek için tekrarlı olarak çok sayıda kayıt yapılmıştır.

SONUÇ ve ÖNERİLER

Performansı gerçekleştirmede kullanılan yöntem, gecikmelerin etkisini azaltmak için denenen yollarla birlikte, oldukça başarılı olmuştur. Özellikleri rastgele süreçlerle belirlenen ekoların üretilmesi ise, denenebilecek çeşitli olanaklar bakımından bir örnek oluşturmuştur.

Kaynağın daha durağan olduğu kısımlarda eşliklerin çalınma zamanlamalarının tümünün analiz penceresi uzunluğunun yolaçtığı sabit gecikme kadar öne alınması, ayrıca elverişli kısımlarda kayda alıp aktarımlı çalma işlemlerinin kullanımı başarılı sonuç vermiştir.

Eşlik partilerinin zamanlamasını erkene almak, çalınışlarının bağlı olduğu zaman çizgisini kaynakla (solistle) eşzamanlı hale getirebilmekle birlikte, perde “içeriğinin” gecikmeli zaman çizgisine göre oluşması, kaynak seste bu küçük zaman farkı içinde perde değişimi olduğu durumlarda eşlik seslerinin başlangıçlarına bir çarpma sesi (*acciaccatura*) eklenmesi biçiminde bir yan etki oluşturmaktadır. Bu küçük zaman farkları içindeki perde değişimlerini nötralize edecek daha çok adımlı aktarımlar programlanabilecek olsa da, düzensiz durumdan düzenli duruma pürüzsüz geçememe sorununun daha parazitli bir sese yolaçması beklenebilir.

Örnek bestenin özelliklerine yönelik ek bir sonuç olarak, ekoların başlangıçlarının armonik bölge değişimlerinde tetiklenecek şekilde programlanması oldukça iyi sonuç vermiştir. Parça, ilgili bölümde de açıklandığı gibi, tonal merkezin çok değişken, sürekli hareket halinde olduğu bir armonik özellikte olduğu için, rastlantısal bölgede ekoların özellikleri çok hızlı değişmektedir. Armonik bölgelerin daha uzun tutulduğu bir parçada (sözelimi, plato modal (bkz. Miller, 1996) kısımlar içeren bir parça gibi) ekolar için uygulanan rastlantısal değerli parametrelerin bazıları (seçilen ivme ve dinamik kontur tipleri) daha kolay algılanabilirdi. (Bu, daha sonraki başka bir uygulama ile denenecektir.)

Bazı etkili geliştirimler şöyle eklenebilir: Üretilen partilerde ses olması amaçlanan kısımlarda solistin partisinde sessiz anlar bulunmaması gereği şeklindeki kısıtlılığı aşmak için, parçanın planına göre, performans sırasında sessiz kısımlardan önceki uygun kısımlarda kayıtlar alınıp sessiz

kısımlarda diğer partileri üretmek için kullanılması programlanabilir. Ayrıca, oluşturulan her bir parti başka bir hoparlör kanalına ayrılarak belirli sayıda – sanal– vokalistin konum bilgilerinin de verildiği bir simülasyon yapılabilir. Perde öteleme teknikleri üzerinde aralıksız süren araştırmalar, programın ilgili modüllerinde kullanılabilecek daha ileri tekniklerin, eşlik partilerinin oluşumundaki pürüzlerin daha da azaltıldığı seslendirmelere olanak vereceğini göstermektedir.

KAYNAKLAR

- Bernsee, S., 1999. Time Stretching and Pitch Shifting of Audio Signals – An Overview. <http://www.thedspdimension.com/admin/time-pitch-overview/>
- Bora, U., 2010. Müzikte İnsan-Makina Etkileşimi Sergileyen Bir Algoritmik Kompozisyon Örneği: “Piyano Fantezisi”. *İzmir Ulusal Müzik Sempozyumu Bildiriler Kitabı*, 90-97, İzmir. <http://izmirulusalmuziksempozyumu.yasar.edu.tr/ulmusebk.pdf>
- Bora, U., 2003. Bilgisayar Destekli Kompozisyon Yöntemlerinde Yeni Yönelimler. *Atatürk Üniversitesi Güzel Sanatlar Fakültesi SANAT Dergisi*, sayı: 4, 133-145.
- Dorran, D., Lawlor, R. and Coyle, E., 2006. A Comparison of Time Domain Time-Scale Modification Algorithms. *Proceedings of the 120th Convention of the Audio Engineering Society*, Preprint 6674, May, Paris, France.
- Duncan, A. and Rossum, D., 1988. Fundamentals of Pitch Shifting. *Proceedings of the 85th Convention of the Audio Engineering Society*, Preprint 2714, October, Los Angeles, USA.
- Hagparast, A., Penttinen, H. and Välimäki, V., 2007. Real-Time Pitch-Shifting of Musical Signals by a Time-Varying Factor Using Normalized Filtered Correlation Time-Scale Modification (NFC-TSM). *Proceedings of the 10th International Conference on Digital Audio Effects (DAFx-07)*, September 10-15, Bordeaux, France.
- Jehan, T., 2007. Real-time Harmonizer. <http://web.media.mit.edu/~tristan/Projects/harmonizer.html>
- Miller, R., 1996. *Modal Jazz Composition and Harmony, Vol.1*. Advance Music.

- Puckette, M., 2002. Max at Seventeen. *Computer Music Journal*, 26(4), 31-43.
- Rimsky-Korsakov, N., 1964. Steinberg, M. (Ed), Agate, E., (Trans). *Principles of Orchestration*. New York: Dover Publications, Inc.
- Rioul, O. and Vetterli, M., 1991. Wavelets and Signal Processing. *IEEE Signal Processing Magazine*, October, 14-38.
- Rossum, D., 1989. An Analysis of Pitch Shifting Algorithms. *Proceedings of the 87th Convention of the Audio Engineering Society*, Preprint 2843, September, New York, USA.