

# 음악의 대표 선율을 이용한 구조-내용 기반 복합 검색을 위한 XML 스키마 설계

김범수\* 김주성\* 김유성\*\*

\*인하대학교 정보통신대학원 정보통신공학과

\*\*인하대학교 정보통신공학부

e-mail:c2021007@inhavision.inha.ac.kr

## Design of a XML Schema for Structural Content-based Composite Retrieval Using Representative Melodies

Bum-Soo Kim\* Joo-Sung Kim\* Yoo-Sung Kim\*\*

\*Graduate School of Information Technology and Telecommunication, Inha University

\*\*School of Information and Communication Engineering, Inha University

### 요 약

XML이 가지는 다양한 장점을 이용하기 위해, 음악을 XML로 표현하려는 다양한 연구가 계속되고 있다. 그러나, SMDL, MNML, MusicXML과 같은 기존의 연구는 악보상의 정보를 그대로 표현하거나, 전자음악의 표준인 MIDI 포맷과의 상호교환성을 중심으로 연구되어 왔기 때문에 음악에 대해 구조-내용 기반 복합 검색을 하기에는 부적합하다. 이를 해결하기 위해 본 논문에서는 음악의 대표 선율을 이용한 구조-내용 기반 복합 검색을 위한 XML 스키마를 제안한다. 먼저, 본 논문에서는 사용자가 질의할 가능성이 높은 첫 동기, 절정, 일정 유사도 내에서 반복되는 주제 선율을 해당 음악을 대표하는 선율로서 XML 스키마에 표현하였다. 제안한 XML 스키마에 대해 XML 질의 언어를 이용하여 구조-내용 기반 복합 검색시 전체 선율을 검색하지 않고 대표 선율만을 검색하여 사용자의 응답 시간을 빠르게 할 수 있다.

### 1. 서 론

최근, XML이 갖는 다양한 장점을 이용하기 위해 음악을 XML로 표현하려는 연구가 계속되었다. XML은 시스템 독립적이고, 가독성이 좋아 문서를 유지 보수하기 쉽다. 또한, XML 문서는 계층적 구조로서 내용과 구조가 분리되어 있다. 따라서 XML은 마디, 동기, 악절로 구성되어 계층적 구조를 갖는 음악을 표현하기에 적합하다[1]. 그러나, SMDL(Standard Music Description Language), MNML(Musical Notation Markup Language), MusicXML등 XML로 음악을 표현하려는 기존의 다양한 연구들은 악보상의 정보를 그대로 표현하거나, 기존 전자음악의 표준 포맷인 MIDI와의 상호교환성에 집중하고 있다[2,3,4]. 따라서 기존의 연구는 '첫 동기가 절정에 해당되는 음악을 찾아라.'와 같이 음악의 구조와 내용을 함께 이용하는 구조-내용 기반 복합 검색에는 적합하지 않다. 즉, 구조-내용 기반 복합 검색을 위해 악보 전체를 검색하기 때문에 사용자의 응답 시간이 오래 걸린다. 이를 해결하기 위해 본 논문에서는 음악의 대표 선율을 이용한 구조-내용 기반 복합 검색을 위한 XML 스키마를 제안한다. 먼저, 음악에 대해 구조-내용 복합 검색시 사용자가 질의할 가능성이 높은 선율인 첫 동기 선율, 절정 선율, 주제 선율등을 해당 음악을 대표하는 선율[5,6]로서 XML 스키마에 표현되도록 한다. 따라서, 음악에 대해 구조-내용 복합 검색시 선율 전체

를 검색하지 않고, 대표 선율만 검색함으로써 사용자의 응답 시간을 단축시킬 수 있다.

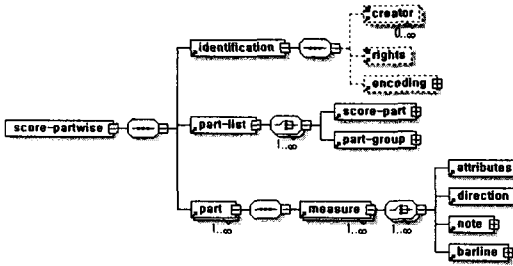
본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴보고, 3장에서는 음악의 대표 선율을 이용한 XML 스키마를 제안한다. 4장에서는 본 논문에서 제안된 XML 스키마를 이용하여 구조-내용 기반 복합 검색 방법을 제시한다. 5장에서는 결론 및 향후 연구 방향을 기술한다.

### 2. 관련 연구

SMDL은 논리적, 시각적, 분석적 도메인 그리고 몸짓에 대한 도메인으로 구분하여 음악을 표현한다[2]. 그러나, SMDL은 SGML의 응용으로 매우 복잡하고, 가독성이 떨어지는 단점이 있다. MNML은 악보상의 기호화 가사들을 가능한 간결하게 표현하기 위해 기술된 언어이다[3]. 그러나, MNML은 음악에 흔히 사용되는 꾸밈음을 지원되지 않는다. MusicXML은 그래픽 환경에서 악보를 편집, 수정할 수 있도록 설계한 악보표현 기법으로 악보를 성부(part)를 기준으로 해석하는 방법과 시간을 기준으로 해석하는 방법을 제공한다[4]. 음악을 XML로 표현하려는 기존의 연구는 '악보의 내용을 그대로 전달하는가?' 또는 'MIDI와 상호교환 가능한가?'에 집중하고 있으며 해당 음악의 전체 선율을 특별한 정보 없이 순차적으로 표현한다. 따라서 음악에 대한 구조-내용 기반 복합

정보 검색시 전체 선율을 순차적으로 검색해야 하는 단점이 있다.

악보는 시간에 따라 변하는 데이터 또는 성부 중심으로 구성된 데이터라 볼 수 있다. 성부 중심의 관점에서 악보를 해석한 MusicXML[4]의 'partwise.dtd'의 구성은 (그림 1)과 같다.



(그림 1) MusicXML의 'partwise.dtd'의 구성도

(그림 1)의 MusicXML의 'partwise.dtd'의 주요 내용을 보면 다음과 같다.

- score-partwise : 최상위 요소를 나타낸다.
- identification : XML문서의 메타정보로서, 'creator'는 속성 'type'에 작곡가, 작사가 등을 표시할 수 있고, 사람의 이름을 값으로 가지며 0개 이상 존재한다. 'rights'는 저작권, 'encoding'은 변환 방법에 대해 저장한다.
- part-list : 성부의 정보로 'score-part'에는 성부이름을 나타내는 'part-name'과 악기 이름을 표시하는 'score-instrument', MIDI 악기의 id 값과 추가 정보를 나타내는 'midi-instrument'가 있고, 성부의 집합 정보를 나타내는 'part-group'이 있다.
- part : 각 성부의 전체 내용으로, 한 마디를 뜻하는 'measure'의 집합이다. 'part'는 속성 'id'로 구분하고, 'measure'는 속성 'number'로 구분한다.
- attributes : 악절의 정보, 'divisions'은 4분음표를 몇 개의 'beat-type'으로 표시할 수 있는가를 표시한다. 'key'는 오도권에 의한 조표(b, #)의 수를 나타내는 'fifths'와, 장단조를 나타내는 'mode'로 표현하고, 박자표는 'time'으로 'beat-type'와 'beats'로 기본음표타입과 기본음표가 한마디에 몇 개인지 표시한다. 'clef'는 음자리표로 'sign'과 'line'이 (G,2), (F,4), (C,3), (TAB,5)값을 가질 수 있다. 그 외에 조옮김, 악기 등을 표시한다.
- direction : 음의 강약이나 꾸밈상태 등 곡의 상태를 표현하는 것으로 곡의 중간이나 시작부분에 위치한다.
- note : 음표의 정보, 'step'은 7음계(C, D, E, F, G, A, B) 중 한 값을 표시한다. 'octave'는 음표가 몇 옥타브에 있는가를 표현하고, 'duration'은 숫자 값으로 'beat-type'값과 곱하면 음표의 길이가 된다. 'type'에서는 2분음표, 4분음표, 8분음표, 16분음표 값을 가지며, 'rest'는 쉼표, 'lyric'은 가사정보, 'dot'은 음표 뒤의 점을 말한다. 'beam'은 잇단음표를 나타내는 것으로 'begin', 'continue', 'end'값을 가질 수 있다.
- barline : 반복이나 종료, 형태와 같은 마디맺음 정보를 갖는다.

따라서, MusicXML은 악보의 내용을 그대로 전달 하면서 해당 음악의 전체 선율을 특별한 정보 없이 순차적으로 표현함을 알 수 있다. XML 표현 예제로 동요 가운데 잘 알려진 '나비나비원나비'를 선택하여 음악을 구성하는 요소 중 독립성을 지닐 수 있는 최소의 단위, 동기로 구분하여 (그림 2)에 표시하였다 [7].



(그림 2) '나비나비원나비'의 악보

사용자가 '첫 동기가 절정에 해당되는 음악을 찾 아라.'와 같이 음악의 구조와 내용을 함께 이용하는 구조-내용 기반 복합 검색시, MusicXML에서는 해당 음악의 첫 동기 선율이 절정에 해당되는지 알아보기

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE score-partwise PUBLIC "-//Recordare//DTD
MusicXML 0.6a Partwise//EN" "partwise.dtd">
<score-partwise>
<identification>
<creator type="composer">Lee, Heung-Ryul</creator>
<encoding><software>Finale for Windows</software>
</encoding>
</identification>
<part-list>
<score-part id="P1">
<part-name>MIDI 1</part-name>
<score-instrument id="P1-I1">
<instrument-name>Acoustic Grand Piano</instrument-name>
</score-instrument>
<midi-instrument id="P1-I1">
<midi-channel>1</midi-channel><midi-program>1</midi-program>
</midi-instrument>
</score-part>
</part-list>
<part id="P1">
<measure number="1">
<attributes>
<divisions>2</divisions>
<key><fifths>0</fifths><mode>major</mode></key>
<time><beats>6</beats><beat-type>8</beat-type></time>
<clef><sign>G</sign><line>2</line></clef>
</attributes>
<note>
<pitch><step>C</step><octave>4</octave></pitch>
<duration>2</duration><voice>1</voice>
<type>quarter</type><stem>up</stem>
</note>
<note>
<pitch><step>F</step><octave>4</octave></pitch>
<duration>1</duration><voice>1</voice>
<type>eighth</type><stem>up</stem>
</note>
<note>
<pitch><step>A</step><octave>4</octave></pitch>
<duration>2</duration><voice>1</voice>
<type>quarter</type><stem>up</stem>
```

위해 전체 선율을 모두 검색하고 각 음들에 대해 음높이 비교를 수행해야 첫 동기가 절정에 해당되는지 판단할 수 있다.

```

</note>
<note>
  <pitch><step>F</step><octave>4</octave></pitch>
  <duration>1</duration><voice>1</voice>
  <type>eighth</type><stem>up</stem>
</note>
</measure>
<measure number="2">
  <note>
    <pitch><step>C</step><octave>4</octave></pitch>
    <duration>1</duration><voice>1</voice>
    <type>eighth</type><stem>up</stem>
    <beam number="1">begin</beam>
  </note>
  <note>
    <pitch><step>F</step><octave>4</octave></pitch>
    <duration>1</duration><voice>1</voice>
    <type>eighth</type><stem>up</stem>
    <beam number="1">continue</beam>
  </note>
  <note>
    <pitch><step>A</step><octave>4</octave></pitch>
    <duration>1</duration><voice>1</voice>
    <type>eighth</type><stem>up</stem>
    <beam number="1">end</beam>
  </note>
  <note>
    <pitch><step>C</step><octave>5</octave></pitch>
    <duration>3</duration><voice>1</voice>
    <type>quarter</type><dot></dot><stem>down</stem>
  </note>
</measure>
<!-- .. 생략 .. -->
</part>
</score-partwise>
    
```

(그림 3) '나비나비흰나비'의 첫 동기 부분

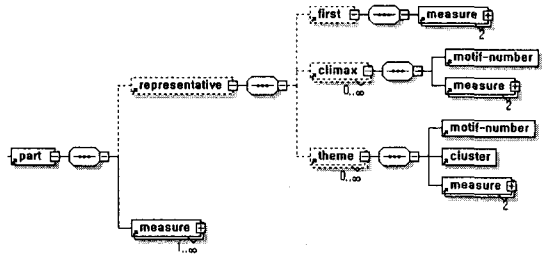
(그림 3)은 (그림 2)를 MusicXML로 나타낸 부분의 첫 동기 부분으로 악보상의 모든 정보가 순차적으로 저장되어 있다. 본 논문에서는 사용자가 질의할 가능성이 높은 첫 동기의 선율, 한 음악의 진행이 모두 향하는 집중되는 부분인 절정선율을 그리고, 일정 유사도 내에서 반복되는 주제 선율을 대표 선율로 인지하고[5,6], 이를 XML 스키마로 표현하여, 음악에 대한 정보 검색시 꼭 전체를 검색하는 것을 피하고 대표 선율을 검색하도록 하였다.

### 3. 음악의 대표 선율을 이용한 구조-내용 기반 복합 검색을 위한 XML 스키마

음악의 대표 선율 정보를 이용하기 위해서 MusicXML의 'partwise.dtd'를 XML 스키마로 수정 확장하였다. 대표 선율을 나타내는 부분은 (그림 4)와 같다.

(그림 4)의 주요 내용은 다음과 같다.

- representative : 'part'의 자식노드로 대표 선율을 나타낸다. 자식노드에는 'first', 'climax', 'theme'가 있다.
- motif-number : 동기번호 값으로 'climax', 'theme'에 필요하다.
- cluster : 일정 유사도 내에 반복되는 선율들의 동기 번호 값이다.
- measure : 선율은 대한 정보로 공간적, 시간적으로 이해할 수 있는 음높이와 길이에 대한 값을 갖는다. 자식 노드인 'note'는 음높이인 'pitch'값과 음길이인 'duration'값만 갖는다.



(그림 4) 제안한 XML 스키마의 대표 선율 부분

대표 선율은 (그림 2)의 악보에서 첫 동기의 선율에 대한 정보, 한 음악의 진행이 모두 향하는 집중되는 부분은 6동기를 절정 선율에 대한 정보[5]로 한다. 그리고, 각 동기별로 음을 시계열 데이터형태로 재구성하여 유사도 행렬을 구하여 유사도 행렬 내에서 유사도가 높은 클러스터를 선택하면 1-3-7, 2-4-8이 각각 후보클러스터가 된다. 클러스터에서 음의 평균길이 변화량과 평균높이 변화량이 중간 값을 가지는 선율을 주제 선율로 선택하게 된다. 1-3-7 클러스터에서는 세 동기가 같은 값을 가지므로 1동기가 선택되고, 2-4-8 클러스터에서는 중간 값을 갖는 8동기가 주제 선율로 선택된다[6]. 대표 선율에 대한 정보를 XML으로 표시하면 다음과 같다.

```

<?xml version="1.0" standalone="no"?>
<!-- .. 생략 .. -->
<part id="PI">
  <representative>
    <first>
      <measure number="1">
        <note>
          <pitch><step>C</step><octave>4</octave></pitch>
          <duration>2</duration></note>
        <note>
          <pitch><step>F</step><octave>4</octave></pitch>
          <duration>1</duration></note>
        <note>
          <pitch><step>A</step><octave>4</octave></pitch>
          <duration>2</duration></note>
        <note>
          <pitch><step>F</step><octave>4</octave></pitch>
          <duration>1</duration></note>
      </measure>
      <measure number="2"><!-- 생략 --></measure>
    </first>
    <climax>
      <motif-number>6</motif-number>
      <measure number="11"><!-- 생략 --></measure>
      <measure number="12"><!-- 생략 --></measure>
    </climax>
    <theme>
      <motif-number>1</motif-number>
      <cluster>1-3-7</cluster>
      <measure number="1"><!-- 생략 --></measure>
      <measure number="2"><!-- 생략 --></measure>
    </theme>
    <theme>
      <motif-number>8</motif-number>
      <cluster>2-4-8</cluster>
      <measure number="15"><!-- 생략 --></measure>
      <measure number="16"><!-- 생략 --></measure>
    </theme>
  </representative>
  <measure number="1">
    <!-- .. 생략 .. -->
  </part>
</score-partwise>
    
```

(그림 5) '나비나비흰나비'의 대표 선율 정보

```

define function Degree(element $thispitch) return integer (
let $step := $thispitch/step
let $octave := integer(string($thispitch/octave))
let $pitchstep :=
if (string($step) = "C") then 0
else if (string($step) = "D") then 1
else if (string($step) = "E") then 2
else if (string($step) = "F") then 3
else if (string($step) = "G") then 4
else if (string($step) = "A") then 5
else if (string($step) = "B") then 6
else 0
return 8 * ($octave) + $pitchstep
)

let $doc := document("나비나비흰나비.xml")
let $representative := $doc/representative

for $pitch in $representative//pitch

let $highnote :=
max(for $pitch in $representative//pitch
return Degree($pitch))
let $lownote :=
min(for $pitch in $representative//pitch
return Degree($pitch))
let $highpitch :=
$representative//pitch[Degree(.) = $highnote]
let $lowpitch :=
$representative//pitch[Degree(.) = $lownote]
let $highmeasure :=
string($highpitch[1]//../@number)
let $lowmeasure :=
string($lowpitch[1]//../@number)
let $highmotif :=
string($highpitch[1]//../..)
let $lowmotif :=
string($lowpitch[1]//../..)

where ($highnote - $lownote) < 7
and $highmotif = $lowmotif

return
<result>
<low-note>{$lowpitch[1]}
<measure-number>{$lowmeasure}
</measure-number>
</low-note>
<high-note>{$highpitch[1]}
<measure-number>{$highmeasure}
</measure-number>
</high-note>
<motif>{$highmotif}</motif>
</result>
    
```

(그림 6) XQuery를 이용한 Q<sub>1</sub> 질의

```

<measure-number>11</measure-number>
</low-note>
<high-note>
<pitch><step>D</step><octave>5</octave></pitch>
<measure-number>11</measure-number>
</high-note>
<motif>6</motif>
</result>
    
```

(그림 7) Q<sub>1</sub> 질의의 결과

4. 구조-내용 기반 복합 검색

본 장에서는 제안된 스키마를 이용하여 '첫 동기가 절정에 해당되는 음악을 찾아라.'와 같이 음악의 구조와 내용을 함께 이용하는 구조-내용 기반 복합 검색을 보인다.

- Q<sub>1</sub>: 음높이 변화가 6도를 넘지 않는 대표 선율의 최고음과 최저음은 어디에 있는가?
- Q<sub>2</sub>: 첫 동기가 절정에 해당되는 음악인가?

질의 결과는 다음과 같다.

```

<?xml version="1.0"?>
<result>
<low-note>
<pitch><step>F</step><octave>4</octave></pitch>
    
```

Q<sub>1</sub>의 질의와 같이 사용자가 질의할 가능성이 높은 선율의 정보를 이용할 시에는 검색대상이 제한된다. 질의 Q<sub>2</sub>를 (그림 6)과 같이 XQuery로 변환하여 질의하면 '나비나비흰나비'의 경우 대표 선율에 등록된 절정 선율이 첫 동기와 같으므로 전체 선율을 검색하지 않아도 된다.

제안된 스키마에서는 XML 질의 언어를 이용하여 구조-내용 기반 복합 검색시 전체 선율을 검색하지 않고 대표 선율만을 검색하여 사용자의 응답 시간을 빠르게 함을 알 수 있다.

5. 결론 및 향후연구

본 논문에서는 음악을 사용자가 질의할 가능성이 높은 대표 선율을 이용해 구조-내용 기반의 복합 검색을 위한 XML 스키마로 표현하여, 음악에 대한 정보를 검색시 성능의 향상을 유도했으며, XML 질의 언어를 이용해 구조-내용 기반 복합 검색 방법을 제시하였다. 향후 연구는 표준화된 음악적 XML 표현방법과 본 논문에서 제시한 검색 방법을 데이터베이스에 적용하여 디지털 음악에 대한 정보검색시스템, 저작권시스템, 상거래시스템으로 확장하는 것이다.

참고문헌

- [1] P.Roland, "XML4MIR: Extensible Markup Language for Music Information Retrieval"
- [2] SMDL : Standard Music Description Language, ISO/IEC DIS 10743:1995
- [3] P.C. Yih Wei, K. Narayan, "Syntax Specification for the Musical Notation Markup Language to be used for Internet Music Transfer and Archiving"
- [4] Michael Good, "MusicXML : An Internet-Friendly Format for Sheet Music"
- [5] 구경이, 임상혁, 이재현, 김유성, "음악 데이터베이스에서 내용 기반 검색을 위한 대표 선율의 추출 및 색인", 한국정보처리학회 논문지
- [6] 하진석, 구경이, 박재현, 김유성, "내용 기반 음악정보 검색을 위한 선율의 시계열 데이터 변환을 이용한 주제 선율 색인 구성", 한국정보처리학회 논문지, 10권 3호, 2003
- [7] 박세광, "음악대사전", 세광음악출판사, 1996