

함수적 변환을 이용한 SMIL2.0 문서의 검증

김경덕[○]

위덕대학교 컴퓨터공학부

kdkim@uu.ac.kr

Verification of SMIL 2.0 Document using a Functional Transformation

Kyungdeok Kim[○]

Division of Computer Engineering, Uiduk University

요 약

본 논문에서는 다양한 상호작용을 가지는 멀티미디어 문서인 SMIL 2.0 문서를 함수적 변환을 이용하여 문서의 검증방법을 제안한다. SMIL 2.0 문서는 기존 SMIL 1.0보다 더 많은 시간 관계의 기술 방법을 제공함으로써 문서 작성의 융통성은 높아지지만 상호작용에 따른 잠재적인 모순의 발생 가능성은 더욱 높다. 특히 비결정행위에 따른 미디어 객체간 상호 참조 및 이벤트 참조에 따라 발생할 수 있는 모순 검증이 필요하다. 제안하는 검증 방법은 SMIL 2.0 문서를 함수적 표현으로 변환하고 객체간 상호참조를 탐색하여 모순 유무를 나타내는 검증 프로토타입이다. 적용 예로는 상호작용을 가지는 멀티미디어 문서의 검증이다.

1. 서 론

상호작용 가능한 SMIL 문서는 웹상에서 분산된 다양한 미디어 객체들의 복합체와 사용자간 상호작용을 제공한다[1, 2]. SMIL 2.0 문서는 기존 SMIL 1.0 문서보다 더욱 융통성 있는 시간관계 모델을 제공하지만, 다양한 융통성에 의하여 문서를 저작할 때 모순된 시간 관계 기술 가능성이 더욱 높다. 이러한 모순된 시간 관계를 가진 SMIL 문서는 저작 의도와 다른 프레젠테이션을 제공함으로써, 모순을 해결하기 위한 기존 연구들은 주로 SMIL 1.0 문서의 시간 관계 모순을 해결하기 위한 연구들이다. 그러나 더 많은 시간 관계 제공하는 SMIL 2.0 문서의 모순성의 검증, 진단, 스케줄링 방법에 대한 연구는 아직 미흡하며, 본 논문에서는 SMIL 2.0 문서의 검증에 대한 방법을 제안한다.

기존 검증 방법으로는 페트리넷, 페트리넷 응용, LOTOS 등의 주로 형식적인 모델을 이용하고 있다. 페트리넷 기반 모델은 논리적이고 양적인 시간을 표기할 수 있지만 작성 및 구현이 어렵다[3, 4, 5, 6, 8]. 페트리넷 응용 모델인 OCPN[4]은 시간 구간을 기반으로 하여 객체들 간의 시간관계를 명확히 나타낼 수 없으며, 이벤트 기반의 관계는 나타내기 어렵다[5]. 특히 배타적인 객체의 재생을 표현하지 못한다. 동적 페트리넷 DPN[6]은 반복적이고 상호작용적인 모델을 기술하기 위하여 스크립트를 페트리 넷의 플레이스(place)에 연관하여 표현하고 있으나, 객체간 상호작용관계가 복잡할수록 문제의 기술과 분석이 쉽지 않으며 스크립트내 공유 변수의 사용으로 해석이 어렵다. LOTOS는 상호작용 기술을 위하여 확장 모델인 RT-LOTOS를 주로 이용하며, 주어진 문서를 접근가능 그래프로 변환하고 비접근 노드의 삭제들

통하여 모순을 검증한다[8]. 하지만 그래프 크기가 문제 규모에 비하여 상대적으로 크고 주어진 그래프로부터 비접근 노드의 탐색이 쉽지 않다. 그러므로 본 논문에서는 기존 시간구간 표현 모델을 확장하여 SMIL 문서의 상호작용 관계를 구조화된 함수 표현으로 변환하여 검증하는 방법을 제안한다.

본 논문의 구성은 다음과 같다. 제 2절에서는 SMIL 문서의 함수적 변환에 대하여 설명하고, 제 3절에서는 SMIL 문서에 적용과 검증 사례를 설명한다. 마지막 제 4절에서 요약 및 향후 연구 방향을 기술한다.

2. SMIL 2.0 문서의 함수적 변환

SMIL 문서를 함수로 변환하기 위하여 본 논문에서는 기존 시간구간 표현 방법[9]을 확장하여 이용한다. 기존 시간구간 표현 방법은 객체의 프레젠테이션 시간구간을 함수로 표현하여 객체간 시간 관계를 표현한다. 본 논문에서는 시간 구간을 가지는 객체로서 시간과 내부 및 외부 이벤트로 확장하여 다양한 상호작용이 가지는 SMIL 2.0문서를 함수적으로 변환한다.

기존 시간구간 표현에서 객체 a의 시간구간은 식 (1)과 같이 정의하며, t 는 시간이고 a_s , a_e 는 시간 구간의 시작과 종료를 나타낸다.

$$a = \{t \mid a_s \leq t \leq a_e\} = (a_s, a_e) \quad \dots (1)$$

SMIL 문서에 속하는 미디어 객체 M은 비디오, 오디오, 애니메이션, 이미지, 텍스트 등으로 구성된다. 시간구간 표현에서 멀티미디어 문서는 시간 구간 연산자들의 결과이며, 연산자는 입력과 출력으로 시간 구간을 가진다. 즉 $t_{i3} = op(t_{i1}, t_{i2})$, op 는 시간구간 연산자이며, t_{i1} , t_{i2} , t_{i3} 는 시간 구간이다. 사용되는 시간 구간 연산자들의

정의는 다음과 같다. 여기서 $a, b \in M0$ 이고 미디어 객체의 시간 구간이며, 시간구간 연산자로는 $seq()$, $follow()$, $par_begin()$, $par_end()$, $par_min()$, $par_max()$, $equal()$, $alt()$, $loop()$ 가 있으며, 연산자 \mapsto 는 시간구간의 출력을 나타낸다.

$seq(a,b) \mapsto (a_s, b_e)$, $follow(a,b) \mapsto (a_s, b_e)$,
 $par_begin(a,b) \mapsto (a_s, a_e)$, $par_end(a,b) \mapsto (a_s, a_e)$,
 $par_min(a,b) \mapsto (a_s, \min(a_s, a_e))$, $loop(a, \infty) \mapsto (a_s, \infty)$
 $par_max(a,b) \mapsto (a_s, \max(a_e, a_e))$,
 $equal(a,b) \mapsto (a_s, a_e)$, $ident(a,b) \mapsto (a_s, b_e)$,
 $alt((a_1, b_1), \dots, (a_n, b_n)) \mapsto (a_s, b_e)$

$seq(a,b)$ 연산자는 객체 a 의 종료시간이 객체 b 의 시작을 정의한다. $follow(a,b)$ 연산자는 객체 a 의 종료시간이 객체 b 의 시작을 정의한다. $par_min(a,b)$ 연산자는 객체 a 가 객체 b 를 시작하고 둘 중 먼저 재생이 종료하는 객체가 나머지 객체의 종료를 정의한다. $par_max(a,b)$ 연산자는 객체 a 가 객체 b 를 시작하고 둘 중 재생 기간이 큰 객체의 종료시간 전체 재생의 종료를 정의한다. $equal(a,b)$ 연산자는 객체 b 의 시작과 종료시간이 객체 a 의 시작과 종료에 종속됨을 정의한다. $ident(a,b)$ 연산자는 객체 a 의 시작이 객체 b 를 시작하고 객체 b 의 종료시간이 객체 a 의 종료를 정의한다. $alt((a_1, b_1), \dots, (a_n, b_n))$ 연산자는 a_1, \dots, a_n 의 객체들 중 가장 적은 구간의 객체를 선택하고 그에 종속된 객체 b 의 종료시간을 정의한다. $loop(a, \infty)$ 연산자는 객체 a 의 재생을 무한 반복하고 외부 이벤트에 의하여 종료시간을 정의한다.

이러한 기존 시간 구간 표현에는 이벤트에 의한 상호작용 행위를 기술하는 표현 방법이 미흡하므로, SMIL 문서의 상호작용 행위를 함수적으로 표현하기 위하여 기존 시간 구간 표현을 확장하기 위하여, 기존 시간구간 표현에서 사용하는 객체에 이벤트 객체를 추가하여 정의한다. 확장된 함수적 표현에서 피연산자는 시작과 종료 속성을 가질 수 있으며, 확장 표현에서 객체 a 의 시간구간은 식2와 같이 정의한다.

$$a = (a_sA, a_eB) = \{t | a_sA \leq t \leq a_eB\} \dots (2)$$

여기서, A, B 는 객체 a 의 시작 및 종료 속성을 나타내고, $[c|d]$ 로 기술한다. 기호 $|$ 의 앞부분 c 는 객체 a 의 재생 시작에 관련된 속성을 나타내며, 뒷부분에 있는 d 는 객체 a 의 종료 속성을 나타낸다. 예로서 $seq(a[c|\emptyset], b[\emptyset|d])$ 는 객체 a 가 이벤트 객체 c 로 시작하는 속성을 가지며, 객체 b 는 이벤트 d 객체로 종료됨을 나타낸다. 객체의 재생 시작 및 종료에 대한 속성이 모두 \emptyset 이면, $a = (a_sA, a_eB) = (a_s, a_e)$ 이다. 시작 및 종료 속성과 연산자의 우선순위에서 함수 연산자내 객체의 재생 시작은 속성이 우선순위를 가지며, 재생 종료에서는 속성과 연산자가 동일 순위를 가진다. 객체 a 의 시간구간은 시작 및 종료 속성에 의하여 재생시간이 유동적으로 변화한다. 기존 연산자중 $loop$ 연산자의 정의는 $loop(aA, n) \mapsto (a_sA, a_eB)$ 과 같이 수정한다. 여기서, B 는 객체 a 의 재생 반복을 나타내며, 시간구간은 그 객체의

재생시간의 배수가 되며, $B = [c| \{a_n, d\}]$, $n = 1, \dots, \infty$, c 는 시작 속성이고 종료 속성은 a 의 n 배 재생 종료 n 또는 이벤트 객체 d 이다. 그리고 추가되는 $excl$ 연산자는 $excl(a_{1,n}, A_{1,n}) \mapsto (b_s, c_e)$ 으로 정의한다. 여기서, $a_{1,n}, A_{1,n}$ 는 각각 a_1, \dots, a_n 와 A_1, \dots, A_n 를 의미한다. $excl$ 연산자는 기존 시간구간 모델의 alt 연산자와 유사하나, 시작 속성에 따라 활성화된 객체는 자신의 종료 속성이거나 다른 피연산자 객체가 활성화될 때까지 재생이 지속된다. 즉 $excl$ 연산자는 사용자의 상호작용에 따라 배타적인 연산을 수행하며, n 개의 객체중 선택된 하나의 객체를 재생한다. 여기서, $(b_s, c_e) = \sum follow(a_iA_i, a_jA_j)$, $i, j \in \{1, \dots, n\}$, b, c 는 첫 번째 선택된 객체와 마지막에 선택된 피연산자 객체를 나타내며, $\{b, c, a_i, a_j\} \subset \{a_k | k=1, \dots, n\}$, $i \neq j$ 이고, i, j 의 선택은 사용자의 상호작용에 따라 변화된다.

SMIL 문서의 함수적 표현으로 변환은 문서의 태그 및 속성을 이용하여 확장 정의된 시간 구간 연산자들의 조합으로 변환한다. 변환 예는 그림 1과 같다.

```

<seq>
<par endsync ="a1">
  <audio id ="a1" src ="..."/>
  <video id ="v1" src ="..."
    begin="btn1.activateEvent"
    end="btn2.activateEvent"/>
</par>
<par begin="btn1.activateEvent">
  <audio id="a2" src="..."/>
  <video id="v2" src="..."/>
</par>
</seq>

seq(par_end(a1.endsync,
  par_begin(a1.v1[btn1.click|btn2.click])),
  par_begin(btn3.click, par_max(a2.v2))
)

```

그림 1. SMIL 문서의 함수적 표현

3. SMIL 문서의 검증

SMIL 문서는 확장된 시간구간 표현으로 함수적으로 변환하면 발생 가능한 논리적 및 시간적 모순성의 탐색이 용이하다. SMIL 문서의 모순은 내부적으로 모순된 시간 관계 설정에서 발생하는 논리적인 모순이 있고 외부적으로는 사용자와 상호작용에 의한 이벤트에 의한 시간 관계 설정에서 발생하는 의미적 모순과 시간적 모순이 있다[8]. 내부 객체 관계 설정에서 발생하는 모순은 객체의 재생시간 구간과 객체 참조 관계로부터 모순을 탐색할 수 있으나, 외부 객체 관계 설정에서 발생하는 모순은 관련 이벤트의 발생 시점에 따라 모순 여부가 결정됨으로 탐색이 쉽지 않다.

논리적인 모순은 객체간 참조가 적절하지 못한 경우에 발생하는 것으로, 예로서 이미 재생이 끝난 비활성 객체를 참조할 때 발생한다. 의미적 모순은 예외적인 이벤트에 의하여 발생하는 것으로, 예로서 외부 이벤트의 참조로 재생 시작하는 객체가 이벤트의 발생 시간이 객체 종료 시점이 지난 시점에 발생할 경우에 잘못된 재생이 발생할 수 있다. 시간적 모순은 객체 재생이 진행할 때 발생하는 이벤트에 의하여 발생하며, 예로는 객체의 재생 동안 사용자 상호작용이나 이벤트의 참조가 필요한 객체가 그 이벤트 발생 시점이 부적절하여 재생이 불가능할 때 발생할 수 있다[8]. 이러한 모순은 모두 내부 및 외부 이벤트의 잘못된 참조로 인하여 발생한다. 그러므로 본 논문에서는 SMIL 문서를 함수적 표현으로부터 객체들의 재생 시간구간과 객체간 참조 관계를 이용하여 모순이 발생가능성을 탐색한다. 예로서 다음 SMIL 문장과 같이 병렬 재생관계에서 재생되는 미디어 객체가 상호 참조 이벤트에 의하여 재생될 때 재생 순서에서 모순이 발생한다. 즉, 객체 a와 v중 어느 객체가 먼저 재생될지 알 수가 없거나 재생이 되지 않을 수 있다. 순차 재생관계에서도 이런 상호참조 이벤트가 발생한다면 재생은 불가능하다.

```
<par>
<audio id="a" src="..." begin=id(v)(begin)" />
<video id="v" src="..." begin=id(a)(begin)" />
</par>
```

주어진 문장의 함수적 표현은 $par_begin(a[v_s|\emptyset], v[a_s|\emptyset])$ 이 되며, 주어진 시작속성에 의하여 두 미디어 객체의 시작이 상호참조 이벤트가 의존성을 가짐으로서 병렬 재생에서 모순이 발생한다. 병렬 재생관계에서는 상호참조가 가능하나 재생의 순서가 교착상태에 빠질 수 있으므로 참조에 따른 해결책이나 재생 순서에 수정이 필요하다. 다음 예는 시간상 후순위로 활성화되는 객체를 참조하는 경우로서, 병렬 재생 시에는 문제가 되지 않는다. 아래 순차 재생 예에서 객체 a가 후순위로 재생되는 객체 v의 재생 참조에 의하여 재생된다면 객체 a는 재생 불가능하다.

```
<seq>
<audio id="a" src="..." begin=id(v)(begin)" />
<video id="v" src="..." />
</seq>
```

주어진 예를 함수적으로 표현하면 $seq(a[v_s|\emptyset], v)$ 이며, a와 v의 재생 시간 구간이 0보다 클 경우, a_s 는 v_s 보다 앞설 수 없으므로 순차재생에서 모순이 발생한다.

SMIL 문서를 검증하는 프로시저는 다음과 같으며, 함수적 표현을 이진 트리구조로 나타내어 문서의 모순성을 검증한다.

```
Procedure Verification_of_SMIL(exp)
{input: exp //functional expression of SMIL document
output: existence of inconsistency
transform exp into BT; // BT : binary tree of exp
//  $\alpha$ : nonterminal node,  $\alpha \in BT$ 
// A, B: left and right sub-trees of  $\alpha$ 
//  $\beta, \delta$ : terminal node,  $\beta \in A, \delta \in B$ 
for ( all  $\alpha$  in BT)
if ( $\alpha$  is a "seq" node) {
if ((( $\beta_s, \beta_e$ ) overlaps ( $\delta_s, \delta_e$ ))) or
(( $\beta$  references  $\delta$ ) or (( $\delta$  references  $\beta$ )))
logical inconsistency occurrence;
} else if ( $\alpha$  is a "par_begin", "par_end",
"par_min" or "par_max" node) {
if (!(( $\beta_s, \beta_e$ ) overlaps ( $\delta_s, \delta_e$ ))) and
(( $\beta$  references  $\delta$ ) or (( $\delta$  references  $\beta$ )))
semantic or temporal
inconsistency occurrence;
} else if ( $\alpha$  is an "excl" node) {
//  $\alpha$ 's child node: $\beta_i, \beta_j, i \neq j, i, j = 1, \dots, n$ 
if ( $\beta_i$  references  $\beta_j$ )
logical inconsistency occurrence;
}
}
```

검증 사례를 보이기 위하여 본 논문에서는 SMIL 2.0의 베이직 프로파일을 기반하는 문서를 이용한다. 베이직 프로파일은 SMIL 2.0 언어 프로파일[1, 2]의 부분집합으로 주요 시간 요소로 <seq>, <par>, <excl>을 가지고 있으며, 여기서 주어진 SMIL 문서는 유효하다고 가정한다.

먼저 순차적 흐름에 병렬적 흐름을 내포하고 이벤트를 가진 그림 1의 SMIL 문서는 그림 2와 같은 이진 트리로 변환되며, 여기서 단말 노드는 이벤트 객체 및 미디어 객체를 나타낸다. 그림 2에 검증 프로시저를 적용하면 비단말 노드를 중심으로 좌우 서브트리 내에 상호 참조가 발생하지 않으므로 모순은 발생하지 않는다. 만약 문서에서 a1 객체의 시작이 a2 객체 시작 또는 종료 이벤트를 참조한다면 주어진 문서는 재생이 불가능하다.

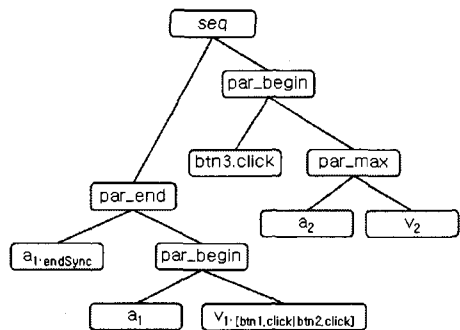


그림 2. 순차적 재생 구조를 가진 경우

다음 SMIL 문서는 배타적 연산의 수행을 가진 예로서 그림 3과 같은 이진 트리구조로 변환된다.

```
<par endsync="select">


<excl id="select">
<text id="t1" src="..." begin="btn1.activeEvent"
dur="25s"/>
<video id="v1" src="..." begin="btn2.activeEvent"/>
</excl>
<audio id="a1" src="..." repeat="indefinite"/>
</par>
```

```
par_end(select.endsync,
  par_begin(btn1,
    par_begin(btn2,
      par_begin
        (excl(select)((t1[btn1.click|∅]),(v1[btn2.click|∅])),
          loop(a1,∞))
        )
      )
    )
  )
)
```

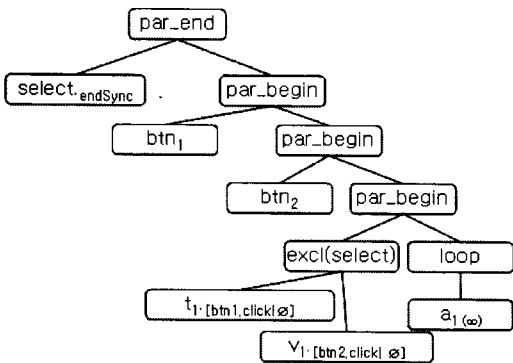


그림 3. 배타적 수행을 가진 경우

그림 3의 배타적 수행을 가진 경우라도 순차 및 병렬 시간 연산자를 기준으로 하여 서브트리 내의 각 객체간 참조관계를 확인함으로써 검증이 용이하다.

SMIL2.0 멀티미디어 문서는 이벤트에 의하여 재생 시간과 순서가 변동되고 이 때 순차 및 병렬 수행관계에서 내부 및 외부 이벤트 참조에 의하여 모순이 발생할 수 있다. 제안한 검증 방법은 SMIL 2.0 문서를 함수적으로 표현하여 객체의 전후 재생 시간과 상호 참조관계를 이용하여 발생 가능한 모순을 재생 전에 탐색하고 수정이 가능하다.

4. 요약

본 논문에서는 SMIL 2.0 문서를 함수적으로 변환하여 문서내 모순을 탐색하는 방법을 제안하였다. 기존 시간 구간 모델을 확장하여 SMIL 문서의 내부 및 외부의 이벤트를 함수로 변환함으로써 검증을 용이하게 하였다. 기존의 검증 방법들도 이벤트 관련한 표현 방법을 제공하고 있으나 사용방법이 복잡하고 표현이 어렵다. 제안한 검증 방법은 이벤트 및 사용자 상호작용을 구조적으로 표현하여 검증 분석이 용이하다.

향후 연구 방향은 제안한 방법을 실시간으로 적용하는 스케줄링 방법의 개발과 일반 프로파일을 적용하는 방법에 대한 연구이다.

참고 문헌

- [1] C. A. Dick, Bulterman, "SMIL 2.0 Part1: Overview, Concepts, and Structure," IEEE Multimedia, pp.82-88, 2001.
- [2] C. A. Dick, Bulterman, "SMIL 2.0 Part2: Examples and Comparisons," IEEE Multimedia, pp.74-84, 2002.
- [1] P. Ackermann, "Direct Manipulation of Temporal Structures in a Multimedia Application Framework," Proc. of ACM Multimedia Conf., pp.51-58, 1994.
- [3] P. Ackermann, "Direct Manipulation of Temporal Structures in a Multimedia Application Framework," Proc. of ACM Multimedia Conf., pp.51-58, 1994.
- [4] M. Vazirgiannis, "Specifying and Authoring Multimedia Scenarios," IEEE Multimedia, pp.24-37, Vol. 6, Issue 3, 1999.
- [5] S. Chung and A. Pereira, "Timed Petri Net Representation of SMIL," IEEE Multimedia, pp.64-72, Jan.-Mar. 2005.
- [6] R. Tan and S. Guan, "A Dynamic Petri Net Model for Iterative and Interactive Distributed Multimedia Presentation," IEEE Transactions on Multimedia, Vol. 7, No. 5, pp.869-879, 2005.
- [7] R. Hamakawa and J. Rekimoto, "Object Composition and Playback Models for Handling Multimedia Data," Proc. of ACM Multimedia Conf. pp.273-281, 1993.
- [8] P.N.M. Sampaio, J.P. Courtiat, "Providing Consistent SMIL 2.0 Document," Proc. of the IEEE Conf. on Multimedia and Expo 2002, Vol. 2. pp. 337-340, 2002
- [9] C. Keramance and A. Duda, "Interval expressions—a functional model for interactive dynamic multimedia presentations," Proc. of the 3rd IEEE Int. Conf. on Multimedia Computing and Systems, pp.283-286, 1996.