# Color Time Petri Net for Interactive Adaptive Multimedia Objects

Ahmed Gomaa, Nabil Adam, Vijayalakshmi Atluri
*Rutgers University*
*Center for Information Management, Integration and Connectivity (CIMIC)*
*{ahgomaa, adam, atluri}@cimic.rutgers.edu*

## Abstract

*A composite multimedia object (cmo) is comprised of different media components such as text, video, audio and image, with a variety of constraints that must be adhered to. The constraints are 1) rendering relationships that comprise the temporal and spatial constraints between different components, 2) behavioral requirements that include the security and fidelity constraints on each component and, 3) user interactions on a set of related media components. Different users have different capabilities (e.g. age), characteristics (e.g. monitor size) and credentials (e.g. subscription to service). Our objective is to author an interactive adaptive cmo that renders itself correctly to different users. Therefore, it is important to guarantee the consistency of the cmo specifications in all possible scenarios. In this paper, we include the user interaction with temporal and spatio-temporal behavior in the specification of the adaptive cmo. We then check the consistency of user interaction specifications by transforming the specifications into a color time Petri net model. We perform a reachability analysis on the Petri net to identify inconsistencies. We then resolve the identified inconsistencies to have a consistent Petri net. A consistent Petri net presents an error-free interactive cmo that can adapt to different users, by guaranteeing that link user interactions are reachable for all eligible users.*

## 1. Introduction

A *composite multimedia object* (*cmo*) is comprised of different media components such as text, video, audio and image, with a variety of constraints that must be adhered to. The constraints include 1) rendering relationships that comprise of temporal and spatial constraints among different components, 2) behavioral requirements that include the security and fidelity constraints on each component. Typically, all these constraints are specified at the time of authoring the *cmo*. Associated with each component is a set of parameters, such as display information, modality, and playback duration. Putting the *cmo* constraints and parameters together leads to a *cmo* plan that describes how the *cmo* will be rendered at the user device.

Different users have different 3Cs: Capabilities (e.g. monitor size), Characteristics (e.g. age) and Credentials (e.g. subscription to services). A user requesting access to a *cmo* may not possess the required credentials to meet the security constraints, or his/her computer system may not possess the desired capabilities to meet the spatial or fidelity constraints. This necessitates modification to the original interactive *cmo* [AAGA03].

In this paper, we include the user interaction with temporal and spatio-temporal behavior in the *cmo* specification to author an adaptive interactive *cmo* that is error-free when rendered at different user devices. We ensure that the adaptation process will not prevent the user to view some components of the *cmo* that he could see if interaction is not included.

A user interaction is either 1) a navigational link (e.g. hyperlink) or 2) an interaction that changes the temporal behavior of a media component (e.g. Fast-forward). User interactions are performed using an interaction device (e.g. mouse, microphone).

To specify a user interaction, we associate one or more interaction devices with one or more media components to perform this interaction. For example, we specify a “mouse-click “ and/or “hotkey” on a “fly-by video” to perform a “fast-forward”. Alternatively, we may specify a “voice command” on a “text” to “ link to” a “map” and “related satellite images”. Specification inconsistencies may arise while specifying user interactions by not including the required user device for a given interaction, or by not specifying the type of interaction allowed on a media place.

Temporal user interactions are user interactions associated with a time window. The time window specifies when the interaction can be performed. The time window is specified in relation to an associated media place. For example, "fast forward" can be applied on a "fly-by video" after 1 minute of the start of the video and 2 minutes before the video ends. The time window can equal to the duration of the associated component. Temporal inconsistencies may occur due to wrong specification of the allowed time window of an interaction.

Single-media spatio-temporal interactions are user interactions associated with a space window. A space window specifies the position where an interaction can occur on the user device's monitor. The space window is specified in relation to an associated media place. For example, a "fast-forward" can be applied to a "fly-by video" when the "fly-by video" "intersects" with the space window region in the user screen. Spatio-temporal inconsistencies may occur due to wrong specification of the allowed space window of an interaction.

### 1.1 The problem statement

The focus of this paper is on addressing the problems of 1) specifying a consistent interactive *cmo* that "automatically" adapts to different user's devices. By consistent we mean consistent specification for temporal interactions, spatio-temporal interactions and interaction devices specifications. 2) Ensuring that the specified interactions will not result in the user not being able to view components that the user would be able to view otherwise.

The paper is organized as follows: section 2 presents preliminaries for our approach. Section 3 discusses the proposed approach. Section 4 presents the related work. Conclusion and future work are discussed in Section 5.

## 2. Preliminaries

In this section, we present the building blocks required for allowing us to model a consistent adaptive interactive *cmo*. We first present the modeling of an adaptive *cmo* by including the temporal, fidelity and security constraints [AAG03], we then include the spatial constraints modeling [GAA04] and thirdly we present the user interaction modeling of a *cmo* [SDKT96].

A color Petri net [J86] is a mathematical tool with graphical representation, where places are circles, tokens a colored dots, transitions are boxes, and arcs are arrows between places and transitions. Each place models an object. A place has an associated color set type determining the value of data that the place may contain. The data is represented in the form of dots named tokens. Tokens are differentiated by colors, which may be arbitrary data values. The transitions are the active part of a system that can change the value of the data. A token can enter a place only if its color is an element in the color set of the place. The arcs are incidence functions that pass the data from places to transitions and from transitions to places. The marking of a place is the existence of tokens in it. The state of the Petri net is the distribution of the tokens in all the places of the net.

### 2.1 Temporal, fidelity and security constraints in color Petri net

In order to effectively represent the temporal, fidelity and security constraints of the multimedia object, [AAGA03] propose to use a color-time Petri net where each place models a media component that is associated with a set of parameters, $\langle da,fd,pd\rangle$ where da is the display attribute, fd is the component fidelity and pd is the playback duration of the component. Unlike traditional color Petri net, they use two types of tokens: a regular token (represented as a filled dot) and a hole (represented as a circle), When a token enters a place, it is said to be available after the duration of the place has elapsed. On the other hand, when a hole enters a place, it becomes available as soon as it enters. As far as the dynamic behavior of the Petri net is concerned, tokens and holes behave the same way.

### 2.2 Spatial Constraints in color Petri net

[GAA04] includes the spatial constraints of the multimedia object in the color Petri net presented in [AAG03]. They present the spatial relations as transitions, where a spatial transition $ST_i$ has only one input place and one output place [PYY03]. A spatial transition $ST_i$ fires token type s-token from a media place $MP_i$ to an address place $AP_i$. $ST_i$ specifies the spatial relation between two media places $MP_i$ and $MP_j$. $ST_i$ does not consume any tokens from $MP_i$. The address place AP following the spatial transition $ST_i$ buffers the spatial token values. The spatial tokens values include the starting address of the following media place and space availability to the following media place.

A graphical illustration of the building block for the model with spatial constraints is shown in figure 1, where $MP_1$, $MP_2$ are media places that accept the set of colors $\langle T, S, I\rangle$, AP is an address place that accepts the color "s-token" and DP is a delay place that accepts all colors. F(VF) is a value function that passes the remaining monitor space to AP. $ST_i$ is spatial transitions that describes an “above” relation. $ST_i$ fires upon arrival of any token in $MP_1$. Assume that the initial marking of $MP_1$ is ⟨T, S, IH⟩ where IH is a hole. $ST_i$ fires immediately. AP buffers the available space if any for $MP_2$.

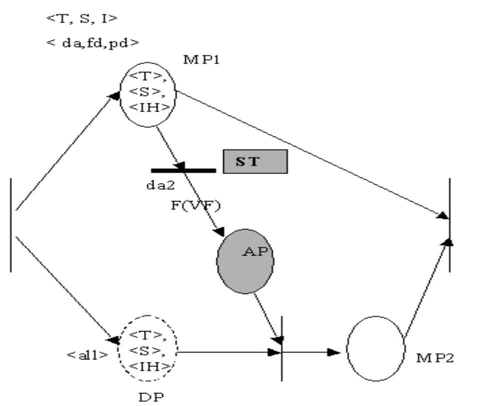


**Figure 1. Spatial constraints in Petri net**

### 2.3 User interaction in time Petri net

[SDKT96] models user interactions in a time Petri net that map hyperstories to Petri net. A hyperstory consists of connected stories and programs using asynchronous events. An event is asynchronous if its occurrence and time of occurrence are unpredictable. The story consists of related multimedia components The hyperstories are linked by two types of flow relationships, branch or call, branching is to terminate the current story and start a new story, while calling is to activate a new story and put the current story on hold, upon the termination of the new story, the original story become active again where it was left off. The presented time Petri Net models timed user interaction for decision-making and to support preemptive resumption, which allows a call to return to where it was left off at its completion.

## 3. Proposed Approach

The major steps of our proposed approach are to (1) define the user interaction specification for the cmo, (2) develop a conceptual model using Color time Petri net (MMCTPN), (3) map the user interaction specification to MMCTPN, and (4) analyze MMCTPN by using the model analysis techniques to identify the user interaction specification inconsistencies.

### 3.1 The user interaction specification

There are two types of user interactions (IT), video-like interaction ($IT_c$) and link interactions ($IT_r$).

$IT_c$ is a video-like interaction that affects the temporal behavior of the media place. $IT_c$ interactions are play, stop or forward, Rewind. “Play” runs the clock value at the normal speed: $fd(C_i) =\{1\}$: such that 1 is the normal clock speed of the media component $C_i$. “Stop” interaction stops the clock value of the media component: fd $(C_i) =\{0\}$. Forward and Rewind interactions defines the duration to be skipped from the media component playback duration, where $fd(C_i) = \{t\}$, such that t is the time duration skipped from $C_i$ playback duration. Rewind interaction defines the duration to be returned from the media place already played duration, where $fd(C_i) = \{-t\}$.

$IT_r$ are link interactions that link two or more components. $IT_r = ((\{C_i\}, L, \{C_j\}), [Resume\vee Reset]$, where $\{C_i\} \wedge \{C_j\} \in C)$. That is, there is a link between two sets of media components where the link either resets or resumes the clock value of the destination components. For example, while browsing a cmo, the user initiates an interaction on an interactive image. The interactive image links to another section of the cmo. The link either resets that section starting time to its beginning, or resumes that section and activates it from where it previously stopped if visited earlier.

User interactions IT= ($IT_c$, $IT_r$) allow two behaviors of interactions: Temporal user interactions and Spatio-temporal user interactions.

Temporal user interactions associate with the user interaction a time window that specifies when the user interaction is allowed. Spatio-temporal interactions associate with them a space window that specifies the location on the monitor where the user interaction is allowed.

**Definition 1:** For a temporal user interaction, a time window $w[x_i,y_i]$ is associated with an active media component $C_i$, where $x_i$ is the lower bound of the time window, $y_i$ is the upper bound of the time window and the playback duration of $C_i$ is $D(C_i)$.

**Definition 2:** For a spatio-temporal user interaction, a space window $SW_i = [(x_s,y_s), h, w]$ is associated with an active media component $C_i$ with display attributes $DA(C_i)$, where $(x_s,y_s)$ is the starting address of the $SW_i$, h is the height and w is the width of $SW_i$. The display attributes $DA(C_i) = [(x_c,y_c), h_c, w_c]$ where $(x_c,y_c)$ is the starting address of $C_i$, $h_c$ is the height and $w_c$ is the width of $C_i$.

A user interaction may have the temporal and spatio-temporal interactions. That is the user interaction is associated with time window and space windows. Such a user interaction can only be active at the specified time window while the specified component intersects with the space window.

The user uses an interaction device at time T to apply temporal interaction IT on one or more components of the cmo. Let IT(Ci) denote the set of all possible user interactions operation for component $C_i$, and $ID(C_i)$ denotes the set of all possible interaction devices for component $C_i$. Where IT= $\{IT_l, . . IT_k\}$ be a set of all distinct user interaction operators in a cmo and ID = $\{ID_l, . . ID_k\}$ be a set of all-distinct interaction devices allowed in a cmo. Each user S possesses a set of interaction devices denoted as $S_D$. When a user S initiates a temporal interaction IT on $C_i$, the user interaction devices $S_D$ are checked against the interaction devices allowed.

**Definition3**: Let $IT(C_i)$ denotes a distinct user interaction $IT_i$ on a media component $C_i$ where $IT(C_i) = \{S_D\} \subset \{ID(C_i)\}$ such that the user interaction is only preformed if the user have the required interaction devices associated with the media component

A temporal user interaction $IT_r$ is inconsistent if the allowed time of interaction falls after the active media component finished its rendering duration.

**Definition 4**: A temporal user interaction is said to be inconsistent if either of the following is true: (1) there exist a lower bound $x_i$ such that $x_i > D(C_i)$ (2) there exist an upper bound $y_i$ such that $y_i > D(C_i)$.

A spatio-temporal user interaction is inconsistent if the allowed window space of interaction falls outside the displayed area of the active media component.

**Definition 5:** A spatio-temporal user interaction is said to be inconsistent if $SW_i \cup DA(C_i) = \phi$

**Definition6**: A user interaction $IT(C_i)$ is said to be inconsistent if (1) $ID(C_i) = \phi$, such that there is no specified user interaction device on a user interaction, or (2) for IT(Ci), $(C_i) = \phi$, such that there is no component $C_i$ associated with the user interaction .

## 3.2 Developing a conceptual model using Color time Petri net- (MMCTPN).

In this section, we build on the work of [AAG03] and [GAA04] by representing the user interaction in Petri net along with the temporal, spatial, security and fidelity constraints. We include all the cmo constraints in one uniform model to allow the authoring of interactive adaptable cmo. We use [SDKT96] methodology in modeling user interaction in our uniform model. The difference between our work and [SDKT96] is that we present the notion of spatio-temporal interaction in Petri net multimedia modeling, we also present allowable interaction devices such as keyboard, mouse and microphone to perform an interaction and how their presences affect the rendering of the cmo. We are using color Petri nets, as it is essential to reduce the modeling complexity. For example, instead of having a single transition for every allowable interaction, we add colors to the transitions for modeling the different interactions

The following definition states that there exist four types of places, a media place (MP) that represent the component, an address place (AP), that buffers the spatial information for the synchronous components, a delay place (DP) that help in prioritizing the firing of spatial transitions, a null place (NP) that connects any two subsequent temporal transitions where at least one media place exist between them and (UP) is a universal place that accepts all types of interaction devices and requires the minimum modality allowed by an implementation language with no security, fidelity or spatial constraints . Each place is associated with a color set (i.e., denoted as C (p)), which specifies the set of allowable colors of the tokens {token, s-token, I-token, holes) to enter the place. Address place, delay places and null places accept all colors with no time duration. (UP) includes a set of token that is minimally required for a multimedia player. There exist a set of output arcs that passes the color tokens to and from the transitions in a predefined time or space window. The set of input arcs either reset the component clock or resume from its previous state. The clock value $D(p_i)$ is used to map the state of a place. The current clock value represents how the place has been active so far from the beginning of its activation. While a place keeps a token, (marked), the clock of the place keeps ticking.

Given a transition t, we denote its input and output set of places by denoted •T and denoted T•, There are four types of transitions T, temporal (TT), Spatial (ST), Interaction (IT) and universal (UT). The dynamic behavior of the MMCTPN can be described in terms of state that represents the distribution of tokens in the net.

An initial state is represented by a set of colored tokens in the source place. The verification of the *cmo* plan starts by firing the output transition of the initial place. The marking of the MMCTPN is changed according to the transitions firing rules.

Let MMCTPN denotes Multimedia Color Time Petri net, where MMCTPN = (P, T, H, A, ∑, D, S0) and
$P = \{p_1, p_2, \ldots p_n\}$ is a finite set of places
$T = \{t_1, t_2, \ldots t_n\}$ is a finite set of transitions,
where $P \cap T = \emptyset$ and $P \cup T \neq \emptyset$.
H =There exists a set of types = {token, s-token, I-token, hole} and a type function H, such that
$\forall y \in m(p), H(y) \in$ {token, s-token, I-token, hole}
A: A→ {IA, OA} an arc is either input or output
∑ = a finite set of colors, color sets ∑ = {σ1, s2,. .},
D = a duration function, D: MP → I* is a mapping from media places to clock values
SP: P → {MP, AP, DP, NP, UP} identifies a place is a Media Place, an Address Place, a Delay Place, a Null place or a universal place.
C = a color function such that $C(p) \subseteq \sum$ and
$\forall y \in m(p), C(y) \in \sum$,
N is the arc function such that: $\forall f(p; t); f(t; p) \in F$; there is an associated set of colors Σ with each arc, where each place has the same color function.
IA: IA → {Reset, Resume} identifies an input arc either resets the clock value of a place or resumes it
W = OA→ [l*], where I* is the set of non-negative integers and W represents temporal window of a user interaction on the output arc
SW = OA→ [l*], where I* is the set of non-negative integers and SW represents the space window of a user interaction on the output arc
T: T → {ST, TT, IT, UT} identifies a transition is either temporal, spatial, interaction or universal
IT: IT → $\{IT_c, IT_r\}$ identifies the video-like interaction and the link interaction.
Q is a transition function such that $\forall f(IT_c) \in F$, there is an associated set of colors ∑with each transition, where each color specify a type of interaction
$S_0$: initial state.

A state of MMCTPN is defined as a tuple State $S_0 = (M_0, \theta)$, for the token distribution in the net at time $Z_0$ Let the state of the system be $S_0 = (M_0, \theta)$. Where $M_0$ denotes <T,S,I> and θ denotes the preserved clock values of the places visited on the net. T is the token marking, S is the s-token marking, I is the I-token marking and θ is the clock value marking. Please note that the place clock value is different than the place duration. The state of MMCTPN is advanced at each transition firing. The $j^{th}$ state of MMCTPN is represented by form of [$< MT_i >$, $< MS_i; >$, $< MI_i >$, $< \theta_i >$]. In this paper we present the firing rules for the user interaction transitions and universal transitions. We use the same firing rules for temporal transitions and spatial transitions as presented in [AAG03] and [GAA04].

Interaction transition firing IT: The interaction tokens {I-token} include all user available interaction devices in one complex color. Any place that required an interaction accepts the I-token and only consumes the interaction device specified for this place. There are two types of user interaction transitions: IT= $(IT_c, IT_r)$. (1) $IT_c$ are video like transitions that change the clock value θ of the media place $MP_i$, those transitions do not consume the media place tokens. (2) $IT_r$ are transitions that serve as a link to another place or set of places.

Given an Interaction transition $IT_c$:

- An interaction transition $IT_c$ is said to be available if there exist in any of its input places an interaction token.
- An interaction transition $IT_c$ is said not to be available if in all input places there is no interaction token or there exist an input place for $IT_c$ with interaction tokens and there exist a token or s-token from type hole.
  - An available $IT_c$ fires if the user initiates a transition on a media place with interaction device.
  - If there is a time /space window on the arc initiating the transition, the time /space window must be enabled.
  - A fired $IT_c$ transition only changes the clock value of the place θ.

Given an Interaction transition $IT_r$:

- An interaction transition $IT_r$ is said to be available if there is in any input place an interaction token.
- An interaction transition $IT_r$ is said not to be available if in all input places there is no interaction token or there exist an input place for $IT_r$ with interaction tokens and there exist a token or s-token from type hole.
- An available $IT_r$ fires if the user initiates a transition on a media place with interaction device.
- If there is a time /space window on the arc initiating the transition, the time/space window must be enabled.
- Firing an $IT_r$ removes marking from input places to output places based on the arc functions.

Universal transition Uti: $\forall MP_i \in \bullet UT_i$, and $\forall MP_i \in UT_i \bullet$ For any media place $MP_i$ marked with m $(mp_i),\theta_i$

- If $IT_r$ is said to be available then m $(mp_i)$ is said to be not available.
- If $IT_r$is said to be unavailable then m $(mp_i)$ is said to be available

$UT_i$ is said to be enabled if m $(mp_i) \bullet UT_i$ is not available. An enabled $UT_i$ transition fires. Firing an $UT_i$ transition consumes only the interaction token from the input place and reproduces it to the UP place. Please note that interaction token includes the entire user available interaction devices .$UP_i$ marking becomes available. $UP_i$ is a special place that includes tokens with the minimal requirement for an implementation language player.

### 3.3 Mapping user interaction specification to MMCTPN.

In this section, we show how MMCTPN can represent the different types of constraints associated with the *cmo* including the user interaction in a uniform way. Since the interaction, temporal and spatial constraints among the different media components are represented in the Petri net model itself by appropriately connecting the places to the transitions, the basic idea is to present the user interaction devices, modality, spatial information, fidelity and security constraints as color sets associated with the places, each place has a duration. While Tokens present the available resources and information (user 3 Cs), holes present the absence of a resource. Token presents the user capabilities (i.e. age), credentials (i.e. member of medscape.com) and characteristics (i.e. installed drivers), S-tokens present the spatial information, I-token presents the interaction devices and colored holes represent the absence of the user 3 Cs. The time duration associated with a place represents its playback duration where θ present the clock value. The time interval associated with an arc represents its availability with respect to the playback duration of the place.

As an example, if a place is associated with a color set = {image}, and a time duration 10, which indicates the modality required by the component to display represented by this place, and the duration as to how long the component is to be played. This component will be triggered only if a token with color image enters this place. On the other hand, if a hole with the color image enters, the component will not be activated, and therefore, the playback duration of the component is zero, hence will not play. This is the case of dropping a component from the *cmo*.

If a user interaction is specified as a link interaction $IT_r$, we connect all spatially connected media places to this interaction with an output arc such that when a link interaction occur, all playing components stop and free their resources for the clicked link output components. Please note that if an audio media place exists with other spatially related components, we connect the audio place with them in a null spatial relation. Two media places are spatially connected with an Address place AP.

To prevent the dropping of a component such that the component is interactive, we use a universal place and a universal transition associated with each link transition. The universal transition only fires if the interactive component cannot be rendered due to the lack of user 3Cs. The universal place does not have any security, fidelity or spatial constraints and accepts all user interactions. By using SMIL 2.0 as our implementation language, the modality of the universal place is <a href > tag specified in SMIL 1.0. The universal place and universal transition replaces the original place to prevent dropping the interaction. [AAGA03] presents the mapping of places and transitions in Petri net to SMIL.

For all $IT_r$, we create a universal transition UT and create a universal place UP, then we connect all the input media places MP connected to $IT_r$ to transition UT, also we connect the transition UT to the universal place UP and finally we connect the place UP to $IT_r$ and all temporal transitions connected to the original input places MP.

```
Input: input places, link transitions, temporal transitions
Output: spatially connected link transitions with universal place

If ∃ MP_i such that MP_i ∈ • IT_ri DO
     If MP_i ∈ •ST_i //* For places that sends their spatial information to AP *//
  Insert output arc OA [MP_j, IT_ri] such that (MP_i < AP_k < MP_j)
If MP_i ∈  TT_i •∧ AP_i ∈ • TT_i // *For places that receives information from AP*//
 Insert output arc OA [MP_j, IT_ri] such that (MP_j < AP_k < MP_i)
Create UT, UP, OA(MP_i, UT_i), IA(UT_i, UP_i), OA(UP_i, IT_ri), OA(UP_i, TT_i)
```

**Figure 2. Connect link to Universal Place**

Figure 3 presents a graphical illustration of the building block for the user interaction and universal transitions. The interactive image is a place $MP_i$ with clock value $\theta_i$, to activate $MP_i$. When the place clock starts ticking, the associated required color (C) must be included as a token, and at least one of the I-token colors (I1, I2, I3) must be included as a color not hole. The user accessing $MP_i$ does not have C or I3 as input tokens, then place $MP_i$ includes a hole for the color C= (CH), and a hole for the I-token I3= (IH3) where I-token =(I1, I2, IH3). $IT_r$ is a link user-interaction transition. $IT_c$ is a video like user interaction transition. UT is a universal transition and T is a temporal transition. The input arc $a_1$ is a resuming arc (with an filled arrowhead), which resumes the clock value $\theta$ in $p_i$. The input arc $a_2$ is a resetting arc (with an empty arrowhead), which resets the clock to its initial value. $IT_r$ has a time window W =[a,b]. $IT_c$ arc function forwards the duration of the image. Since the place cannot activate the clock value due to the presence of the hole CH, then neither $IT_r$ nor $IT_c$ can be enabled. In addition, $TT_i$ cannot be enabled before the duration of the UP place "Minimal tag link" $\theta_j$ elapses. Only UP will be active and prevent the dropping of the image place link. Please note that firing a UP transition only consumes the interaction token to produce the activation token of UP.

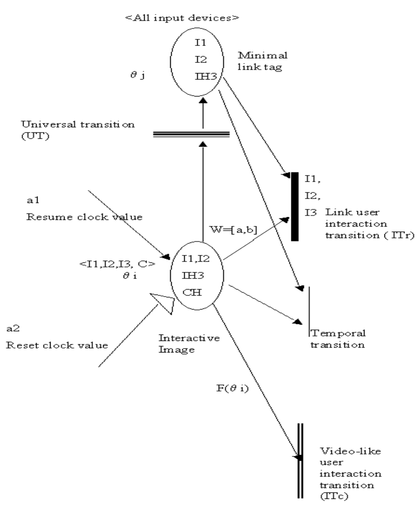


**Figure 3. User interaction and universal place**

## 3.4 Analysis of MMCTPN:

Once the interactive *cmo* is modeled using Petri net, we test for the consistency of the *cmo* specification. We identify temporal inconsistencies in specifying the temporal user interactions in the adaptive *cmo* by traversing the net and comparing each time window with the associated media place and check for their intersection. We also identify the spatio-temporal inconsistencies in specifying the spatio-temporal user interactions in the adaptive *cmo* by traversing the net and comparing each space window with the media place size and check if they intersect. Third, we identify the reachable places and transitions in the net to check the consistency in specifying interaction devices. That is to check the consistency of assigning colors to media places for invoking an interaction. We build on [AH00] approach to remove the timing from the color Petri net to not worry about the temporal aspect of the net for interaction device specification consistency. We extract the temporal attributes from the MMCTPN. This separates a color Petri net (MMCPN) from the MMCTPN. The temporal extraction facilitates testing the consistency of assigned colors to places without worrying about the temporal aspect of the net.

**Definition 7:** Given a MMCTPN extract MMCPN :

1. For each pi, remove D(pi), and
2. For each ti, remove W(ti).

By removing the temporal properties of MMCTPN we simply remove the playback duration of the places and the time intervals associated with arcs.

**3.4.1 Temporal and spatio-temporal inconsistency Analysis.** We traverse the Petri net by firing the transition rules and we check for the consistency of both temporal and spatio-temporal interactions based on their consistency definition. Figure 4 presents the pseudo code.

**Definition 8:** the *cmo* is said to be temporally consistent if the schedulable. That is the duration specified on an output arcs can be satisfied if the temporal duration of a media place D ($MP_i$) $\geq$ the lower bound firing time of an interaction, and D ($MP_i$) $\geq$ the upper bound firing time of an interaction. W[a,b] is the specified lower and upper bound for time interval to fire a transition on the output arc from MPi to a user interaction transition $IT_i$.

**Definition 9:** the *cmo* is said to be spatio-temporally consistent if for a spatio-temporal interactive media place there exist a space window. That is the space specified on an output arcs can intersect the display attributes of a media place MP to fire a user interaction transition $IT_i$ , where MP (starting point, height, Length) $\cap$ space window (starting point, height, Length).

```
Input: MMCTPN, initial marking, final marking
Output: inconsistent temporal and spatio-temporal
interaction

Tunvisited ← T ,  While Tunvisited ≠ 0;
Tvisited ={t_i ∈Tunvisited |m (p_j) ≠ϕ∀p_j∈ •t_i}
// * Get Tlf, Teft firing time and the Tspace of each
Transition where Tlf, Teft  are upper and lower
bounds  and Tspace is the space window *//
  {For each t_i∈Tvisited Teft(t_i)← {λ^l},Tlft(ti)← {λ^u},
Tspace(t_i)←(x,y,h,w) }
{For each t_i ∈Tvisited , fire each transition  }
/*Based on the firing rules of each transition
Tunvisited←Tunvisited-Tvisited,End {while}
Treachable = D (p_i) - Teft (t_i) ∧ D (p_i) - Tlft (t_i)
Sreachable =MP_i(x,y, h, w) ∩ SW_i(x,y,h,w)
If ∃t_i ∈Treachable such that Treachable < 0
T_i is not temporally consistent
If ∃t_i ∈Sreachable such that MP ∩ SP= ϕ
T_i is not spatio-temporally consistent
```

**Figure 4. Find inconsistent temporal and spatio-temporal transitions**

**3.4.2 User interaction inconsistency:** An interactive media place is associated with at least one related transition, such that any media places {MP} that associate any interaction device color {Δ}, is either connected to a video or link user interaction transition IT. To assure that this is the case, we first generate the reachability tree of the Petri net, then 1) we check if there are interactive media places that are not connected to any video like interaction transitions and 2) we also check if there are interactive media places that are not connected to any link interaction transitions. 3) We identify the media places that are not connected to video transitions or link transitions by intersecting of the output of step 1 and 2.

**Definition 10:** Unconnected interactive places are the places that are not connected to any video like transition or any link transition while associated with those places interaction device colors. Such that unconnected interactive places = (ITr1-unconnected array ∩ ITc1-unconnected array)

We generate the reachability tree from the MMCTPN by using Jensen algorithm for color Petri net reachability tree algorithm [J86]. Each node in the resulting tree is a matrix with [j] rows representing the places of the net and [i]columns representing the color associated with each place.

In video like transitions, we check the input and output nodes in the reachability tree for an interactive media place with same interaction device color token before and after the transition firing. The transition accepts the same interaction device color. We remove the interaction device color token from the media place and the transition. The interaction color tokens that remains in the input place are added to the unconnected array for video like transitions. By pruning the media places with their related transitions, we identify the media places with ability to interact (interaction device tokens) but without user interaction transitions.

In link interaction transition, we check for all the media places with different marking before and after the transition firing, then we remove all the related tokens from input places and related transition from the reachability tree. The interaction color tokens that remains in the input place are added to the unconnected array for link transitions.

Since some places can be interactive and related to video while other places are simply links to other places, the intersected between the outputs of the two types of interaction transitions are the interactive media places that are not video like or links, thus inconsistent.

```
Input: R, a reachability tree with n nodes
Output: list of places in nodes that does not have
related user interaction video-like transitions

For each node R_k
        If R_k is a leaf {Process node( ) }
        else { goto child node }
Process node ( )
 {for (ITc = 1 to n)
//where n is the number of video like trans.  from node i
   {for ( P = 1 to m)
//where m is the number of places in node I
    {for R [i] [j] ∈ {Δ}
      {for each ψ_i in Δ
 // ψ_i denotes the allowable interaction devices for P
        { if •MP_i  = MP_i• ∨ IT_ci ∈ ψi
// Transition accepts the same interaction device
       remove ψ_ITci  from IT_ci , ψ_i from  R [i] [j]
        else add (R[i][j], /ψ_I) to ITc_unconnected
}}}}};
// *Add the media place and the input device that can
interact with it *//
```

**Figure 5. - Inconsistent video-like interaction**

```
Input: R, a reachability tree with n nodes
Output: list of places in nodes that does not have
related user interaction link transitions

For each node R_k
        If R_k is a leaf {Process node( ) }
        else { goto child node }
Process node ( )
 {for (ITr = 1 to n)
//where n is the number of link trans.  from node i
  {for ( P = 1 to m)
//where m is the number of places in node I
   {for R [i] [j] ∈ {Δ}
     {for each ψ_i in Δ
 // ψ_i denotes the allowable interaction devices for P
      { if •MP_i ≠ MP_i• ∨ ITr_i ∈ ψi
// Transition accepts the same interaction device
      remove ψ_ITri   from ITr_i , ψ_i from   R [i] [j]
       else add (R[i][j], /ψ_I) to ITr_unconnected
}}}}};
// *Add the media place and the input device that can
interact with it *//
```

**Figure 6. - Inconsistent link interaction**

**Example:** In the following example, interactive image $P_1$, with duration = 3 minutes, accepts three interaction devices: voice, keyboard and mouse. A video-like interaction device changes the duration of the image while a link interaction transition direct the *cmo* to a video $P_3$. The link interaction time window is [2-3]. If no interaction is used, the interactive image remains for its duration. The interactive image has a security constraint, such that only subscriber at myweb.com can view the image. Parallel with this image there is a text file $P_2$ in French (F) with duration of 3 minutes. The following video with duration = 4 has no requirements other than the modality =mpeg1. Each component size is 100*100 pixels. AP is buffering the display information of $P_1$ to $P_2$, while ST states the spatial relation $P_1$ above $P_2$.

A French-speaking user accessing the *cmo* that has a keyboard and a mouse as interactive devices and is not a subscriber of myweb.com, her system have a monitor of resolution 640*480 pixel, and she has the codec for mpeg1 and can view texts and images.

By inserting the user 3Cs as tokens in the *cmo*, we generate the reachability tree of the MMCPN after removing the time

**T1 fires.** Tokens moves from the initial place $P_0$ to $P_1$, DP and Null. UP is a universal place that includes all minimum values.

**ST fires.** Tokens move to the address place AP that buffers the display information of $P_1$ and the space availability to $P_2$. ST does not consume any tokens from $P_1$.

**T2 fires.** Tokens move to $P_2$. T2 fires due to AP and SP availability with an available space to display. $P_1$ and Null still remain unchanged.

**UT fires.** Interaction tokens move to UP. Since both $IT_c$ and $IT_r$ are unavailable, they cannot fire

**$IT_r$ enabled.** Since there exist one available input place with interaction token and without a time interval.

**T3 enabled** since all input places have available tokens.

If the user interacts on UP during its allowed duration, then the video will display, else the *cmo* will end.

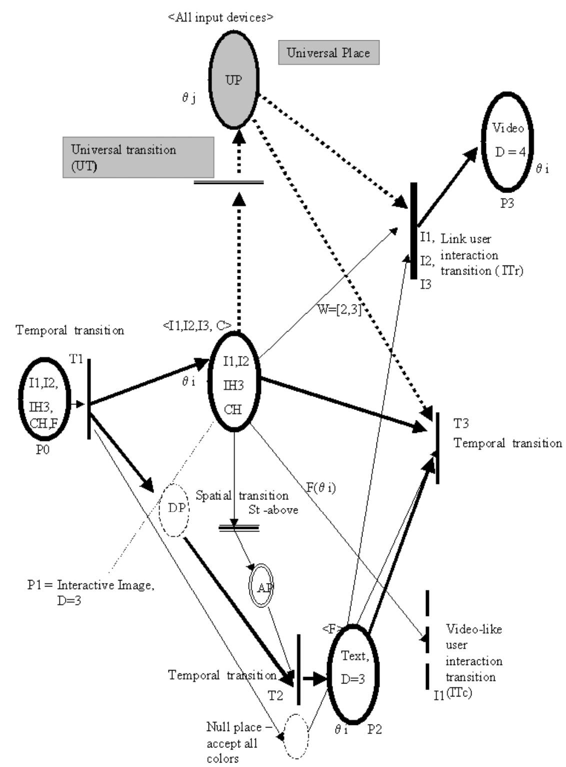


**Figure 7. – MMCTPN example**

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | T, V,I | 640*480 | Hole | 640*480 | I1,I2, H3 | F |
| P1 | | | | | | |
| DP | | | | | | |
| AP | | | | | | |
| P2 | | | | | | |
| UP | Min | Min | Min | Min | | |
| P3 | | | | | | |
| Null | | | | | | |

T1

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | | | | | | |
| P1 | I | 640*480 | Hole | 640*480 | I1,I2, IH3 | |
| DP | T, V,I | 640*480 | null | 640*480 | I1,I2, IH3 | |
| AP | | | | | | |
| P2 | | | | | | |
| UP | Min | Min | Min | Min | | |
| P3 | | | | | | |
| Null | T, V,I | 640*480 | null | 640*480 | I1,I2, IH3 | F |

ST

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | | | | | | |
| P1 | I | 640*480 | Hole | 640*480 | I1,I2, H3 | |
| DP | T, V,I | 640*480 | null | 640*480 | I1,I2, H3 | |
| AP | | | | Availa | | |
| P2 | | | | | | |
| UP | Min | Min | Min | Min | | |
| P3 | | | | | | |
| Null | T, V,I | 640*480 | null | 640*480 | I1,I2, H3 | F |

T2

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | | | | | | |
| P1 | I | 640*480 | Hole | 640*480 | I1, I2,H3 | |
| DP | | | | | | |
| AP | | | | | | |
| P2 | T | 640*480 | null | Avail | | F |
| UP | Min | Min | Min | Min | | |
| P3 | | | | | | |
| Null | T, V,I | 640*480 | null | 640*480 | I1, I2,H3 | F |

UT

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | | | | | | |
| P1 | I | 640*480 | Hole | 640*480 | | |
| DP | | | | | | |
| AP | | | | | | |
| P2 | T | 640*480 | Null | 640*480 | | F |
| UP | Min | Min | Min | Min | I1, I2 | |
| P3 | | | | | | |
| Null | T, V,I | 640*480 | Null | 640*480 | I1, I2 | F |

ITr

| | Mod. | Fid. | Sec. | Spa. | Inter | L |
|---|---|---|---|---|---|---|
| P0 | | | | | | |
| P1 | | | | | | |
| DP | | | | | | |
| AP | | | | | | |
| P2 | | | | | | |
| UP | | | | | | |
| P3 | V | | | 640*480 | | |
| Nul | | | | | | |

**Figure 8. – Reachability tree**

## 4. Related work

[LG90] introduce multimedia presentation using Petri net. [DS93] introduce the network jitter in the multimedia presentation in a formal framework. [WQG94] model sub multimedia objects in Petri net to model multimedia presentations in distributed environment. [PR94] present Dynamic Timed Petri Nets to support the specification of user interactions such as skip, reserve, freeze, restart, and scale presentation speed, slower or faster. It allows user-defined interrupts to preempt the Petri net execution sequence and modify the time duration associated with the preempted Petri net process. [GYY98] present a distributed object composition Petri net model (DOCPN) to handle the synchronization and interaction for distributed multimedia systems. Their model supports user interaction and a resulting schedule of current media flow. [AAAB01] present the first Petri net model for adapted multimedia presentations where they consider dropping the components without using the analysis power of Petri net. [AAGA03] present an extension to [AAAB01] by adding the security constraints into an enhanced Petri net model and used the Petri net as the tool for dropping the unmatched components. [GAA04] present the spatial constraint in a color Petri net model along the other constraints and present adaptation algorithm to adjust optional spatial constraints to adapt to the user device. [VTS98] introduce the concept of a uniform spatio-temporal multimedia modeling, [PYY03] present the first Petri net multimedia model that presents both spatial and temporal constraints. Temporal consistency in multimedia presentations has been extensively addressed using different approaches as [SDKT96], [GYY98], and [PR94], [LK95], [MBSTV00], [SLC01] to generate a temporal consistent interactive multimedia object.

## 5. Conclusion and future work

In this paper, we present a formal model for user interaction in adaptive composite multimedia objects.
Namely, the paper maps user interactions (video like and links) and their associated temporal and spatio-temporal behavior along the specified interaction devices to a Color Time Petri net model that adapts to different users. The model enables the following. 1) Visualize the user interactions along the other constraints in one model with minimizing the graph explosion problem of the net by using colors for transitions and places. 2) Identify and resolve temporal and spatio-temporal inconsistencies for user interaction specification 3) Identify inconsistencies in the specified interaction devices to perform user

interaction. More analysis can be performed, for example, deadlocks checking can be performed by P-invariant analysis to check for siphons. A spatio-temporal interaction can be extended to be relational, that is, it is specified on a certain spatial relation that exists between two media places. For example a "sensitive data link" can only be present if the "fly-by video" and the "digital map" are displayed at the same time, leaving room to the author to create adaptive *cmo* for different viewers. Future research includes identifying other potential inconsistencies in the spatio-temporal user interaction and the spatial interaction. In addition changing the modalities of different components in the *cmo* to adapt to different devices and insuring temporal consistency when the change of modality is introduced.

## References


[ J86] Kurt Jensen: Coloured Petri Nets. Advances in Petri Nets 1986: 248-299

[AAAB01] N. R. Adam, V. Atluri, I. Adiwijaya, and S. Banerjee. A Dynamic ManifestationApproach for Providing Universal Access to Digital Library Objects. IEEE TKDE 13(4) 2001.

[AAGA03] V Atluri N Adam A Gomaa I Adiwijaya, Self-manifestation of composite multimedia objects to satisfy security constraints, ACM SAC 2003.

[AH00] V. Atluri and W-K. Huang, ``A Petri Net Based Safety Analysis of Workflow Authorization Models," *Journal of Computer Security,* Volume 8, Issue 2/3, 2000

[DS93] Diaz, M. and Senac, S Time Stream Petri Nets: A Model for Multimedia Streams Synchronization. Proceedings of the First International Conference on Multi-Media Modeling, 1993.

[GAA04] Ahmed Gomaa and Vijay Atluri and Nabil Adam. Composite Multimedia Object adaptation to user device and related resolution strategies. CIMIC Technical report UA-0403

[GYY98] Sheng-Uei Guan, Hsiao-Yeh Yu, Jen-Shun Yang A Prioritized Petri Net Model and Its Application in Distributed Multimedia Systems. IEEE transaction on computers 47( 4). 1998

[LG90] T.D.C. Little and A. Ghafoor. Synchronization and Storage Models for Multimedia Objects. IEEE J. Selected Areas in Communication, 8(3) 1990.

[LK95] Layaida, N. , Keramane, C. Maintaining temporal consistency of multi-media documents. In: ACM Workshop on Effective Abstractions in Multimedia, San Francisco, 1995

[PR94] Prabhakaran, B., Raghavan, S.V. .Synchronization Models for Multimedia Presentation With User Participation. ACM/Springer-Verlag Journal of Multimedia Systems 2 : 2, 1994.

[PYY03] Hsu, Ping-Yu; Chang, Yuan-Bin; Chen, Yen-Liang, STRPN: A Petri-Net Approach for Modeling Spatial-Temporal Relations between Moving Multimedia Objects. IEEE Transactions on Software Engineering, Volume 29, Issue 1, pages 63-76. 2003.

[SDKT96] Junehwa Song, Yurdaer N. Doganata, Michelle Y. Kim, Asser N. Tantawi. Modeling Timed User-Interactions in Multimedia Documents.International Conference on Multimedia Computing and Systems (Multimedia '96)

[SLC01] P. N. M. Sampaio , C. Lohr , J. P. Courtiat, An integrated environment for the presentation of consistent SMIL 2.0 documents, Proceedings of the 2001 ACM Symposium on Document engineering, 2001

[VTS98]M. Vazirgiannis, Y. Theodoridis, and T. Sellis, "Spatio-Temporal Composition and Indexing for Large MultimediaApplications,"ACM/Springer-Verlag Multimedia System J., vol. 6, no. 4, 1998.

[WQG94] Woo, R., Qazi, N. U., & Ghafoor, A. A synchronization framework for communication of pre-orchestrated multimedia information. IEEE Network, Special Issue on Distributed Applications for Communications, 8(1) 1994.Conf.,pages 320 -40,Nov 1996.