

Modeling Interactive Distributed Multimedia Applications

Sheng-Uei Guan

National University of Singapore, Singapore

INTRODUCTION

In recent years, researchers have tried to extend Petri net to model multimedia. The focus of the research flows from the synchronization of multimedia without user interactions, to interactions in distributed environments (Bastian, 1994; Blakowski, 1996; Diaz, 1993; Guan, 1998; Huang, 1998; Huang, 1996; Little, 1990; Nicolaou, 1990; Prabhakaran, 1993; Prabhat, 1996; Qazi, 1993; Woo, 1994). The issues that concern us are the flexibility and compactness of the model. Petri net extensions have been developed to facilitate user interactions (UI) in distributed environments; however, they require sophisticated pre-planning to lay out detailed schedule changes. In this article, we introduce a Reconfigurable Petri Net (RPN).

An RPN is comprised of a novel mechanism called a modifier (f), which can modify an existing mechanism (e.g., arc, place, token, transition, etc.) of the net. A modifier embraces controllability and programmability into the Petri net and enhances the real-time adaptive modeling power. This development allows an RPN to have a greater modeling power over other extended Petri nets. The article introduces both the model and theory for RPN and a simulation to show that RPN is feasible.

BACKGROUND

Little (1990) has proposed the use of Object Composition Petri Net (OCPN) to model temporal relations between media data in multimedia presentation. The OCPN model has a good expressive power for temporal synchronization. However, it lacks power to deal with user interactions and distributed environments. Extended Object Composition Petri Net (XOCPN), proposed by Woo, Qazi, and Ghafoor (1993), is an improved version of OCPN with the

power to model distributed applications, but it does not handle user interactions.

The lack of power in OCPN to deal with user interactions has led to the development of an enhanced OCPN model, Dynamic Timed Petri Net (DTPN) proposed by Prabhakaran and Raghavan (1993). DTPN provides the ability for users to activate operations like skip, reverse, freeze, restart, and scaling the speed of presentation.

Guan (1998) has proposed DOCPN to overcome the limitations of the original OCPN and XOCPN. DOCPN extends OCPN to a distributed environment using a new mechanism known as prioritized Petri nets (P-nets) together with global clock and user interaction control. Guan and Lim (2002) later proposed another extended Petri net: Enhanced Prioritized Petri Net (EP-net), an upgraded version of P-net. It has a Premature/Late Arriving Token Handler (PLATH) to handle late and/or premature tokens (locked tokens forced to unlock). Moreover, EP-net has another feature: a dynamic arc that simplifies and improves the flexibility of designing interactive systems.

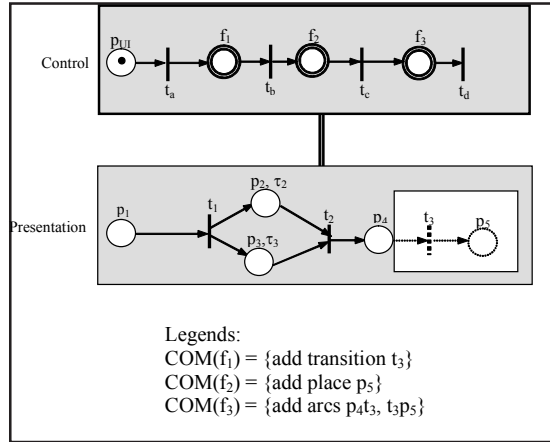
None of the above-mentioned Petri nets have controllability and programmability built in as RPN has offered to Petri net, neither do they have the ability to model a presentation on the fly and simulate real-time adaptive application.

RECONFIGURABLE PETRI NET

Definitions

RPN consists of two entities: control and presentation layers. Each entity is represented as a rectangle. These two layers can be joined together by a link (denoted by a double line). A link represents necessary interactions between the control layer and the presentation layer. Note that multiple presentation

Figure 1. RPN: A simple example

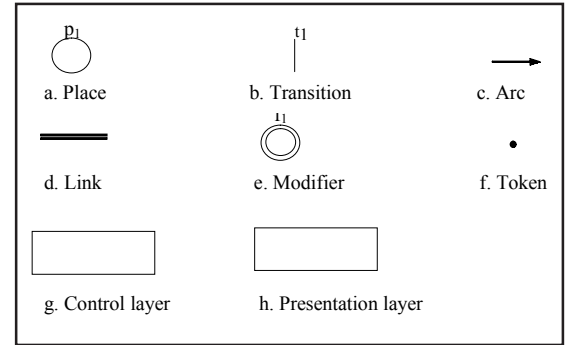


and control layers could exist in a model. An example of RPN is shown in Figure 1. Initially, there are no mechanisms (e.g., t_3 and p_5) inside the white box of the presentation layer. After the activation of the modifiers (e.g., f_1 , f_2 , and f_3) in the control layer, the mechanisms inside the white box are created. First, the control layer as shown in Figure 1 starts with a token in p_{U1} , transition t_a is enabled and fires. The token is removed from p_{U1} and created at modifier f_1 . Upon the token arriving at modifier f_1 , transition t_3 is then created in the presentation layer. Transition t_b is enabled and fires only if a token is present at f_1 and the transition t_3 is created. After transition t_b is enabled and fires, the token in modifier f_1 is removed and created at modifier f_2 . Upon the token arriving at modifier f_2 , the place p_5 is created in the presentation layer. Next, transition t_c is enabled and fires. The token in modifier f_2 is removed and created at modifier f_3 . Upon the token arriving at modifier f_3 , the arcs p_4t_3 and t_3p_5 are created in the presentation layer. Finally, transition t_d is enabled and fires. The token is removed from modifier f_3 . Therefore, we have shown how a RPN works. In the following, we explain the definitions related to RPN.

Definition: Control and Presentation Layers

The structure of an unmarked layer in RPN is a six-tuple, $S = \{T, P, A, D, L, COM\}$. For marked RPN's layer, the definition of the structure becomes a seven-tuple, $S = \{T, P, A, D, L, COM, M\}$. Refer to the

Figure 2. RPN graphic representations



structures S as mentioned above, where $P \cap T = \emptyset$. A complete RPN net may consist of zero or more control layers and one or more presentation layers.

$T = \{t_1, t_2, t_3, \dots, t_m\}$ is a finite set of transitions where $m > 0$.

$P = \{p_1, p_2, p_3, \dots, p_i, f_4, f_5, f_6, \dots, f_k\}$ is a finite set of places and/or modifiers where i and $k > 0$.

$COM: f_a \rightarrow \{com_1, com_2, com_3, \dots, com_z\}$ is a mapping from the set of modifiers to the commands (as defined in Table 1) where a and $z > 0$.

$A: \{P \times T\} \cup \{T \times P\}$ is a set of arcs representing the flow relation.

$M: P \rightarrow I^+$, $I^+ = \{0, 1, 2, \dots\}$ is a mapping from the set of places or modifiers to the integer numbers, representing a marking of a net.

$D: p_b \rightarrow R^+$ is a mapping from the set of places to the non-negative real numbers, representing the presentation intervals or the durations for the resources concerned where $b > 0$.

$L = \{c_x \text{ or } p_x\}$ indicates whether an entity is a control layer c_x or presentation layer p_x where $x > 0$.

The set of graphical symbols for RPN are demonstrated in Figures 2a to 2h. A classic place is shown in Figure 2a, which represents a resource (e.g., audio, video playback, etc.). If the place is associated with duration (D), it indicates the interval of the resource to be consumed. Figure 2b displays a transition, which represents a synchronization point in a presentation. In Figure 2c, an arc is demonstrated which represents a flowing relation in a presentation. Then, the links as shown in Figure 2d establish connections linking two

5 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/modeling-interactive-distributed-multimedia-applications/17312

Related Content

A Social Media Recommender System

Giancarlo Sperli, Flora Amato, Fabio Mercorio, Mario Mezzanzanica, Vincenzo Moscato and Antonio Picariello (2018). *International Journal of Multimedia Data Engineering and Management* (pp. 36-50).

www.irma-international.org/article/a-social-media-recommender-system/196248

Dimension Reduction Using Image Transform for Content-Based Feature Extraction

Sourav De, Madhumita Singha, Komal Kumari, Ritika Selot and Akshat Gupta (2018). *Feature Dimension Reduction for Content-Based Image Identification* (pp. 26-40).

www.irma-international.org/chapter/dimension-reduction-using-image-transform-for-content-based-feature-extraction/207226

Supporting Online Communities with Technological Infrastructures

Laura Anna Ripamonti (2005). *Encyclopedia of Multimedia Technology and Networking* (pp. 937-944).

www.irma-international.org/chapter/supporting-online-communities-technological-infrastructures/17350

Enhancing E-Commerce through Sticky Virtual Communities

Sumeet Gupta (2009). *Encyclopedia of Multimedia Technology and Networking, Second Edition* (pp. 477-481).

www.irma-international.org/chapter/enhancing-commerce-through-sticky-virtual/17438

FaceTimeMap: Multi-Level Bitmap Index for Temporal Querying of Faces in Videos

Buddha Shrestha, Haeyong Chung and Ramazan S. Aygün (2019). *International Journal of Multimedia Data Engineering and Management* (pp. 37-59).

www.irma-international.org/article/facetimemap/233863