

## KIỂM TRA RÀNG BUỘC TOÀN VỆN THỜI GIAN TRONG CƠ SỞ DỮ LIỆU THỜI GIAN

### CHECKING TEMPORAL INTEGRITY CONSTRAINTS IN TEMPORAL DATABASES

*Phạm Văn Chung, Nguyễn Thành Sơn*  
Trường ĐH. Công nghiệp, ĐHSPTK Tp HCM

*Dương Tuấn Anh*  
Trường Đại học Bách khoa – ĐHQG Tp HCM

#### TÓM TẮT

*Bài báo trình bày một phương pháp kiểm tra ràng buộc toàn vẹn thời gian trong cơ sở dữ liệu thời gian bằng cách dùng đồ thị chuyển trạng thái. Thứ nhất, bài báo đưa ra cách dùng những câu có-khuôn-mẫu để đặc tả ràng buộc toàn vẹn thời gian và nêu cách chuyển chúng thành những đồ thị chuyển trạng thái. Thứ hai, đề xuất một giải thuật để kiểm tra ràng buộc toàn vẹn thời gian bằng cách dùng đồ thị chuyển trạng thái mà những lối đi trên đó cho phép có đỉnh lặp lại. Cuối cùng, bài báo trình bày một thí dụ kiểm tra ràng buộc toàn vẹn thời gian trong một cơ sở dữ liệu bệnh viện*

#### ABSTRACT

*This paper presents a method for checking temporary integrity constraints in real world temporal databases by using transition graphs. First, it advocates the usage of patterned-sentences to express temporal integrity constraints and gives a method to transform patterned-sentences to transition graphs. Second, this work puts forward an algorithm for checking temporary integrity constraints by utilizing the transition graphs in which the paths are allowed to include repeated vertices. Finally, the paper illustrate a case study that performs checking temporary integrity constraints in a temporal clinical database.*

#### I. DẪN NHẬP

Trong bài báo này, chúng tôi nghiên cứu làm thế nào để kiểm tra ràng buộc toàn vẹn của dữ liệu theo thời gian (RBTVTG). Dựa trên hướng tiếp cận của Gertz và Lipeck [4], chúng tôi mở rộng khung thức của họ đã đề xuất để kiểm tra RBTVTG trong trường hợp đối tượng trong cơ sở dữ liệu chuyển trạng thái trở lại những trạng thái cũ.

Kiểm tra ràng buộc toàn vẹn nhằm duy trì ngữ nghĩa và tính đúng đắn của dữ liệu theo thời gian bất cứ khi nào có cập nhật dữ liệu trong cơ sở dữ liệu thời gian (CSDLTG). Do vậy, cần tìm một mô hình toán học để thực hiện công việc này với chi phí thấp và dễ dàng cài đặt. Có một số hướng tiếp cận kiểm tra RBTVTG nổi bật như là: phương pháp History-less checking của Chomicki [1], [2], và phương pháp dùng đồ thị chuyển trạng thái (*transition graph*) của Gertz và Lipeck [4].

Bằng cách dùng những câu có-khuôn-mẫu (*patterned-sentence*) để đặc tả RBTVTG và sau đó chuyển nó thành đồ thị chuyển trạng

thái. Đồ thị chuyển trạng thái là đồ thị có hướng, biểu thị chu kỳ sống của những đối tượng trong cơ sở dữ liệu (*Life cycles of database objects*). Đỉnh của đồ thị biểu thị trạng thái mà những đối tượng có thể đi đến, cạnh biểu thị điều kiện mà những đối tượng phải thoả khi chuyển đến đỉnh khác theo hướng của cạnh. Kiểm tra RBTVTG cần thực hiện hai phần: kiểm tra tính toàn vẹn của dữ liệu theo thời gian, và kiểm tra ràng buộc về sự thay đổi trạng thái của đối tượng theo một qui luật của tổ chức.

Phần còn lại của bài báo được tổ chức như sau: Phần 2, lược yếu một số hướng tiếp cận kiểm tra RBTVTG. Phần 3, trình bày tóm tắt: làm thế nào để đặc tả một RBTVTG bằng những câu có-khuôn-mẫu và chuyển nó thành đồ thị chuyển trạng thái tương ứng. Phần 4, trình bày một số những khái niệm quan trọng và những yêu cầu liên quan đến đồ thị chuyển trạng thái. Phần 5, mô tả những phương pháp kiểm tra RBTVTG. Phần 6, hiện thực việc kiểm tra RBTVTG trong phần 5. Cuối cùng là kết luận và hướng nghiên cứu tiếp tục.

## II. MỘT SỐ HƯỚNG TIẾP CẬN ĐI TRƯỚC

**Hướng tiếp cận của Chomicki.** Chomicki [1], [2] đưa ra phương pháp gọi là *history-less checking* để kiểm tra RBTVTG. Phương pháp này khi kiểm tra RBTVTG, không cần kiểm tra trên toàn bộ dữ liệu, mà chỉ cần kiểm tra trên một số dữ liệu cần thiết được đặt trong những quan hệ hỗ trợ (*auxiliary relation*). Những quan hệ hỗ trợ được sinh ra một cách tự động khi RBTVTG được định nghĩa, điều này giảm được chi phí trong kiểm tra RBTVTG và đây là điểm nổi bật về cách tiếp cận RBTVTG của Chomicki.

**Hướng tiếp cận của Gertz và Lipeck.** Những tác giả này [4],[5],[6],[7] dùng lô-gic thời gian (temporal logic) như ngôn ngữ đặc tả RBTVTG và từ những công thức lô-gic thời gian, nó được chuyển thành đồ thị chuyển trạng thái. Mỗi đồ thị chuyển trạng thái có thể mô tả chu trình sống của những đối tượng trong cơ sở dữ liệu. Những đỉnh của đồ thị biểu thị những thông tin tối thiểu cần cho việc kiểm tra RBTVTG của đối tượng trong cơ sở dữ liệu.

Hướng tiếp cận này đưa ra một khung thức tổng quát [4], dùng đồ thị chuyển trạng thái để kiểm tra RBTVTG. Tuy nhiên, những ứng dụng trong thực tế, đối tượng chuyển trạng thái phức tạp hơn, ví dụ như những ứng dụng trong theo dõi, điều trị bệnh trong cơ sở dữ liệu bệnh viện. Do vậy, cần có sự mở rộng thêm hướng tiếp cận này, để giải quyết những trường hợp phức tạp đó.

## III. CÂU CÓ-KHUÔN-MẪU VÀ ĐỒ THỊ CHUYỂN TRẠNG THÁI.

RBTVTG có thể được đặc tả bằng công thức lô-gic thời gian [3]. Bài báo này đề xuất câu có-khuôn-mẫu để có thể dễ dàng đặc tả RBTVTG hơn là dùng lô-gic. Chi tiết hơn về cách đặc tả này có thể xem trong [7]. Ví dụ: xét một RBTVTG (*phỏng theo ví dụ trong [4]*) được phát biểu như:

“ Trước khi một nhân viên (*Employee*) trở thành thành viên của phòng *Headquarters*, anh ta phải làm việc tại phòng *Administrative*, và một khi đã làm việc tại đây thì anh ta luôn ở vị trí này cho đến khi anh ta nghỉ việc”

RBTVTG này có thể trình bày bằng câu có-khuôn-mẫu [7] như trong hình 1. Đối tượng *e* trong hình 1, tham khảo đến đối tượng có kiểu *Employee*, và có thể chuyển đặc tả này thành đồ thị như ở hình 2. Đồ thị này mô tả những trạng thái mà trong chu kỳ sống của đối tượng, nó có thể chuyển đến. Tập đỉnh của đồ thị được đánh nhãn là 0, 1 và 2 ứng với những trạng thái  $s_0$ ,  $s_1$  và  $s_2$  và những nhãn trên cạnh ứng với những phát biểu của  $l_1, l_2, l_3$ , và  $l_4$  trong hình 1

Object is in first state  $s_0$  with  $l_2$  moves to  $s_1$ ,

when it is in  $s_0$  with  $l_1$  moves to  $s_0$ ,

when it is in  $s_1$  with  $l_3$  moves to  $s_1$ ,

when it is in  $s_1$  with  $l_4$  moves to  $s_2$ ,

when it is in  $s_2$  with  $l_4$  moves to  $s_2$ ,

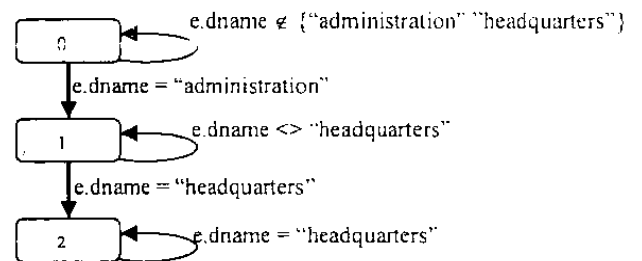
where  $l_1 \equiv e \notin \{“administration”, “headquarters”\}$ ,

$l_2 \equiv e = “administration”$ ,

$l_3 \equiv e \diamond “headquarters”$ ,

$l_4 \equiv e = “headquarters”$ ;

Hình 1. Đặc tả RBTVTG bằng câu có-khuôn-mẫu



Hình 2. Đồ thị chuyển trạng thái sinh ra từ câu có-khuôn-mẫu

## IV. MỘT SỐ ĐỊNH NGHĨA TRÊN ĐỒ THỊ CHUYỂN TRẠNG THÁI

Để kiểm tra RBTVTG trên những trạng thái mà đối tượng có thể đi qua trên đồ thị chuyển trạng thái. Trong phần này, cần định nghĩa một số khái niệm quan trọng.

### 4.1 Những trạng thái được lặp lại trong dãy chuyển trạng thái

Xem đồ thị hình 3, nó có nhiều chu trình giữa một số đỉnh. Kiểm tra RBTVTG trong trường hợp này, bài báo đề xuất một phương pháp là dùng một số nguyên dương  $t$ , gọi là giá trị tần suất. Phương pháp được thể hiện như:

lần đầu tiên, đối tượng ở trạng thái khởi đầu, giá trị tần suất  $t$  được gán giá trị là 0. Đối tượng đến một trạng thái mới  $s_i$ , nhưng  $s_i$  là trạng thái mà đối tượng đã đi đến trong một lần trước đây, thì giá trị  $t$  tăng thêm 1. Mọi trường hợp khác thì giá trị  $t$  không đổi.

**4.2 Định nghĩa 4.1** (Đồ thị chuyển trạng thái)

Một đồ thị chuyển trạng thái là đồ thị có hướng gồm những thành phần:  $V$  là một tập hữu hạn những đỉnh (trạng thái),  $L$  là tập gồm những nhãn hay những mệnh đề điều kiện,  $\delta: V \times L \rightarrow V$  là hàm chuyển trạng thái,  $v_0 \in V$  là đỉnh khởi đầu, và  $V_e = V$  là tập đỉnh kết thúc, trong  $V_e$  có thể có một số đỉnh mà tại đó không có cạnh đi ra, gọi chúng là đỉnh  $v_f$ . Trên mỗi cạnh được gán một nhãn là một phần tử trong  $L$ . Mỗi đỉnh phải có ít nhất một cạnh đi vào, và nếu nó có cạnh đi ra thì những cạnh này phải được gán những nhãn khác nhau. Từ một đỉnh bất kỳ (trừ những đỉnh  $v_f$ ) phải có đường đi đến những đỉnh  $v_f$ .

**Ví dụ:** đồ thị chuyển trạng thái  $T$  như trong hình 3 có những thành phần:

$$V = \{s_1, s_2, s_3, s_4, s_5\}, L = \{l_1, l_2, l_3, l_4, l_5, l_6\}.$$

$$v_0 = s_1, V_e = V, v_f = s_5.$$

và hàm chuyển dịch  $\delta$ :

$$\delta(s_1, l_1) = s_2, \delta(s_1, l_2) = s_3, \delta(s_2, l_3) = s_1,$$

$$\delta(s_2, l_4) = s_4, \delta(s_3, l_4) = s_4, \delta(s_4, l_5) = s_1,$$

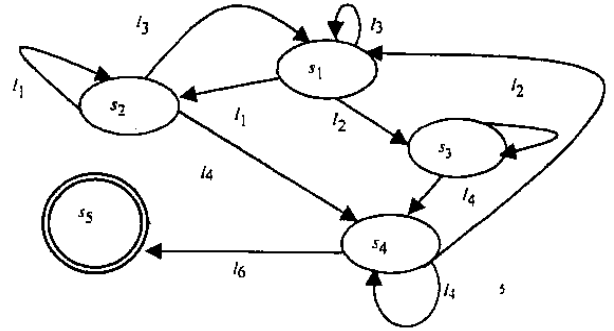
$$\delta(s_4, l_6) = s_5, \dots$$

$s_1$  là trạng thái ban đầu và  $s_5$  là trạng thái kết thúc.

**4.3 Định nghĩa 4.2** (Dãy chuyển trạng thái)

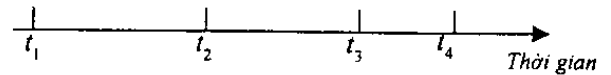
Một dãy có thứ tự gồm một số trạng thái trong đồ thị chuyển trạng thái, đầu dãy là trạng thái khởi đầu (đối tượng có thể chuyển từ trạng thái khởi đầu), thoả những nhãn trên cạnh để đến những trạng thái khác theo thứ tự trong dãy gọi là dãy chuyển trạng thái.

Ví dụ:  $\sigma = (s_1, s_2, s_1, s_3, s_4)$  là dãy chuyển trạng thái trên hình 3.



Hình 3: Đồ thị  $T$ ,  $s_1$  là trạng thái ban đầu

**4.4 Định nghĩa 4.3** (Điều kiện về thời gian)   
Coi hai trạng thái liên nhau  $i$  và  $j$  trong đồ thị chuyển trạng thái (có cạnh đi ra từ  $i$  đến  $j$ ). Giả sử rằng thời gian hợp lệ của hai trạng thái liên nhau của một đối tượng tại  $i$  và  $j$ , hay cùng tại một trạng thái  $i$  là  $[t_1, t_2]$  và  $[t_3, t_4]$ . Nếu  $t_3 \geq t_2$ , thì  $[t_1, t_2]$  và  $[t_3, t_4]$  thoả mãn ràng buộc theo thứ tự thời gian, xem hình 4.

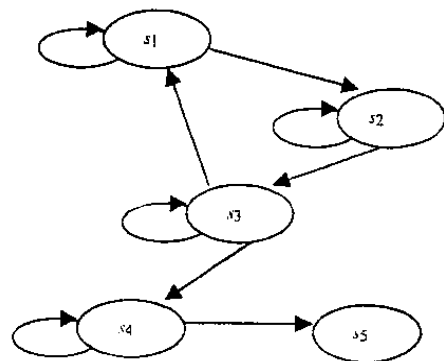


Hình 4: Ràng buộc theo thứ tự thời gian

**4.5 Định nghĩa 4.4** (Ràng buộc qui luật chuyển trạng thái)

Thêm dữ liệu cho đối tượng  $O$  tại trạng thái  $j$  được chấp nhận, nếu thoả một trong những điều: (1)  $O$  đang tồn tại ở trạng thái  $i$ , có cạnh đi ra từ  $i$  đến  $j$ , và thoả mãn nhãn trên cạnh này. (2)  $j$  là trạng thái khởi đầu của đồ thị và đối tượng chưa tồn tại ở bất kỳ trạng thái nào. (3)  $O$  đang tồn tại trong trạng thái  $j$  và  $j$  có ít nhất một đường đi ra.

**Ví dụ:** xem đồ thị chuyển trạng thái trong hình 5.



Hình 5: Đồ thị chuyển trạng thái,  $s_1$  là trạng thái đầu

- Nếu đối tượng đang ở  $s_1$ , thêm dữ liệu tại  $s_4$ , bị từ chối vì không thoả (1), (2) và (3) nhưng thêm ở  $s_1$ , được chấp nhận vì thoả (3).
- Nếu đối tượng chưa tồn tại trong bất kỳ trạng thái nào, yêu cầu thêm nó vào  $s_1$ , thì được chấp nhận, vì thoả (2).
- Nếu đối tượng đã tồn tại trong  $s_5$ , thêm vào nó vào  $s_5$  thì từ chối vì không thoả (1), (2) và (3)

**4.6 Định nghĩa 4.5** (Ràng buộc trên dãy chuyển trạng thái)

Coi  $\sigma = (s_1, s_2, \dots, s_n)$  là một dãy chuyển trạng thái của đối tượng  $O$ . Nếu một cập nhật làm mất đi một trạng thái bất kỳ trong  $\sigma$ , cho dù trạng thái đó được lặp lại trong  $\sigma$ , ngoại trừ  $s_n$ , thì vi phạm ràng buộc trên dãy chuyển trạng thái.

Ví dụ Coi đồ thị trong hình 3, giả sử đối tượng có  $\sigma = (s_1, s_2, s_1, s_3, s_4)$ . Cập nhật làm mất đối tượng ở bất kỳ trạng thái nào trong  $\sigma$ , trừ  $s_4$  là vi phạm ràng buộc trên dãy chuyển trạng thái. Đặc biệt trong trường hợp này,  $\sigma$  có chu trình như đã nói trong phần 4.1, nên cập nhật không cho phép mất  $O$  ở  $s_1$ , trong cả 2 lần  $s_1$  xuất hiện trong  $\sigma$ .

**V. KIỂM TRA RBTVTG TRÊN ĐỒ THỊ CHUYỂN TRẠNG THÁI**

Phần này, trình bày kiểm tra RBTVTG khi có thao tác: thêm, xoá, sửa dữ liệu trên đồ thị chuyển trạng thái có chu trình. Giả sử rằng mỗi đỉnh của đồ thị thì tương ứng với một bảng trong CSDLTG. Các thao tác thêm, xoá, sửa sau được hiểu là thao tác trong cùng một tần suất  $t$ .

**5.1 Thao tác thêm**

Thêm đối tượng vào đỉnh  $j$  của đồ thị chuyển trạng thái, cần kiểm tra thoả mãn định nghĩa 4.3 và 4.4.

**5.2 Thao tác xoá**

Dữ liệu của đối tượng có thể được xoá tại một đỉnh của đồ thị. Khi xoá cần kiểm tra thoả định nghĩa 4.5.

**5.3 Thao tác sửa**

Đối tượng  $O$  đang có giá trị  $x$  ứng với khoảng thời gian  $[O_s, O_e]$  tại đỉnh  $i$ , yêu cầu sửa lại giá trị  $x$  thành là  $y$  ứng với khoảng thời gian cho trước  $[t_1, t_2]$ . Tùy theo giá trị của  $[t_1, t_2]$  mà có thao tác sửa tương ứng như trong bảng 5.3.

Bốn trường hợp	Trước khi sửa R	Sau khi sửa R																		
$O_s$ _____ $O_e$ $t_1$ _____ $t_2$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>O_e</math></td></tr> </table>	A	S	E	x	$O_s$	$O_e$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>t_1^-</math></td></tr> <tr><td>x</td><td><math>t_2^+</math></td><td><math>O_e</math></td></tr> <tr><td>y</td><td><math>t_1</math></td><td><math>t_2</math></td></tr> </table>	A	S	E	x	$O_s$	$t_1^-$	x	$t_2^+$	$O_e$	y	$t_1$	$t_2$
A	S	E																		
x	$O_s$	$O_e$																		
A	S	E																		
x	$O_s$	$t_1^-$																		
x	$t_2^+$	$O_e$																		
y	$t_1$	$t_2$																		
$O_s$ _____ $O_e$ $t_1$ _____ $t_2$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>O_e</math></td></tr> </table>	A	S	E	x	$O_s$	$O_e$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>t_2^+</math></td><td><math>O_e</math></td></tr> <tr><td>y</td><td><math>O_s</math></td><td><math>t_2</math></td></tr> </table>	A	S	E	x	$t_2^+$	$O_e$	y	$O_s$	$t_2$			
A	S	E																		
x	$O_s$	$O_e$																		
A	S	E																		
x	$t_2^+$	$O_e$																		
y	$O_s$	$t_2$																		
$O_s$ _____ $O_e$ $t_1$ _____ $t_2$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>O_e</math></td></tr> </table>	A	S	E	x	$O_s$	$O_e$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>t_1^-</math></td></tr> <tr><td>y</td><td><math>t_1</math></td><td><math>t_2</math></td></tr> </table>	A	S	E	x	$O_s$	$t_1^-$	y	$t_1$	$t_2$			
A	S	E																		
x	$O_s$	$O_e$																		
A	S	E																		
x	$O_s$	$t_1^-$																		
y	$t_1$	$t_2$																		
$O_s$ _____ $O_e$ $t_1$ _____ $t_2$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>x</td><td><math>O_s</math></td><td><math>O_e</math></td></tr> </table>	A	S	E	x	$O_s$	$O_e$	<table border="1"> <tr><td>A</td><td>S</td><td>E</td></tr> <tr><td>y</td><td><math>O_s</math></td><td><math>O_e</math></td></tr> </table>	A	S	E	y	$O_s$	$O_e$						
A	S	E																		
x	$O_s$	$O_e$																		
A	S	E																		
y	$O_s$	$O_e$																		

**Bảng 5.3:** Bốn trường hợp cho sửa dữ liệu vào bảng R, A là thuộc tính cần sửa dữ liệu x thành y, S và E là thuộc tính thời gian.

Ghi chú: trong bảng 5.3, giả sử  $t_1$  là ngày 10/11/2004, thì  $t_1^-$  là ngày 9/11/2004 và  $t_1^+$  là ngày 11/11/2004.

Chúng tôi đã xây dựng những thủ tục kiểm tra theo các thao tác thêm, xoá và sửa như trong các mục 5.2, 5.2 và 5.3, và ứng dụng nó trong phần 6. (có thể xem thêm chi tiết trong [7]).

**VI. HIỆN THỰC KIỂM TRA RBTV**

Xét một cơ sở dữ liệu theo dõi điều trị bệnh nhân ung thư, gồm có một số bảng. Bảng PATIENT chứa những dữ liệu của bệnh nhân gồm những thuộc tính *P-ID*, *Gender*, *DOB* và *Address*, trong đó *P-ID* là mã số của bệnh nhân.

Bảng này có mối quan hệ với những bảng khác (xem hình 6). Trong mỗi bảng ở hình 6, ngoại trừ bảng PATIENT đều có chứa thuộc tính thời gian (*V\_Begin*, *V\_End*), và thuộc tính TIME để ghi tần suất  $t$  ( $t$  đã đề cập đến trong mục 5.2). Những thuộc tính được gạch dưới là khóa chính của bảng).

- 1) PATIENT (P-ID, Gender, DOB, Address)
- 2) UNTREATED ( Prob, V\_Begin, P-ID, Times, V\_End, Vertex\_From, Vertex\_To)
- 3) SURGERY( Prob, V\_Begin, P-ID, Times, V\_End, Vertex\_From, Vertex\_To)
- 4) RADIATION (Prob, V\_Begin, P\_ID, Times, V\_End, Vertex\_From, Vertex\_To)
- 5) CHEMOTHERAPY (Prob, V\_Begin, P\_ID, Times, V\_End, Vertex\_From, Vertex\_To)
- 6) WATCHING (P\_ID, Prob, V\_Begin, Times, V\_End, Vertex\_From, Vertex\_To)
- 7) RECOVER( P\_ID, V\_Begin, V\_End, Times)

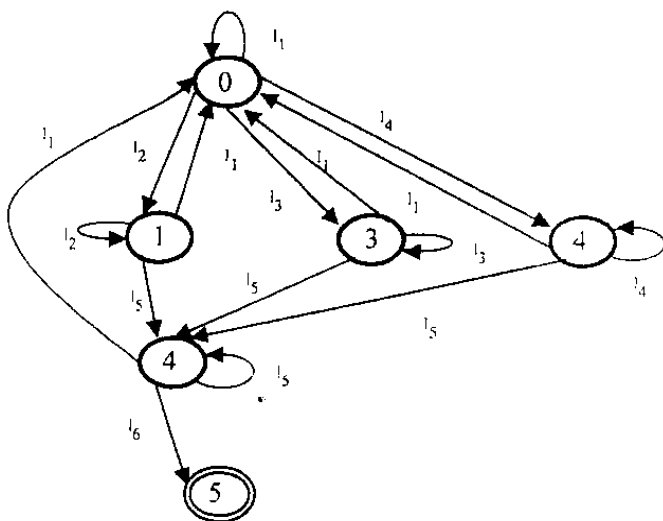
**Hình 6.** Cấu trúc các lược đồ trong cơ sở dữ liệu

Những bảng trong hình 6, được xây dựng từ đồ thị chuyển trạng thái trong hình 7, mô tả RBTVTG trong cơ sở dữ liệu bệnh viện, trong đó đỉnh 0, 1, 2, 3, 4, 5 ứng với các bảng UNTREATED, SURGERY, RADIATION, CHEMOTHERAPY, WATCHING, RECOVER theo dõi bệnh ung thư.

Nội dung của nhãn là:

$l_1 \equiv 'untreated'$ ,  $l_2 \equiv 'surgery'$ ,  $l_3 \equiv 'radiation'$   
 $l_4 \equiv 'chemotherapy'$   $l_5 \equiv 'watching'$ ,  $l_6 \equiv 'recover'$

Đối tượng được nói đến trong cơ sở dữ liệu này có kiểu PATIENT



**Hình 7.** Đồ thị chuyển trạng thái cho cơ sở dữ liệu bệnh viện

• RBTVTG được đặc tả bằng đồ thị chuyển trạng thái

Đồ thị chuyển trạng thái trong hình 7 đặc tả RBTVTG được phát biểu như sau:

*“Lúc đầu, bệnh nhân nhập viện, được đặt trong trạng thái chẩn đoán (untreated), sau đó anh ta có thể được điều trị bằng phương pháp phẫu thuật (surgery) hay xạ trị (radiation) hay hoá trị (chemotherapy). Nếu anh ta được điều trị bằng một trong ba phương pháp trên, anh ta có thể trở lại trạng thái chẩn đoán hay anh ta có thể được theo dõi (watching) trong một khoảng thời gian. Đặc biệt, nếu anh ta đang ở giai đoạn theo dõi, anh ta cũng có thể trở lại trạng thái chẩn đoán hay được kết luận bình phục (recover)”*

RBTVTG trong phát biểu này, được đặc tả bằng câu có-khuôn-mẫu (đã nói đến trong mục 3) và nó được chuyển đổi thành đồ thị chuyển trạng thái (hình 7).

• Tạo lập những bảng lên kết với đồ thị chuyển trạng thái

Ngoài những bảng như trong hình 6, những bảng sau đây được tạo lập một cách tự động khi có đồ thị chuyển trạng thái để liên kết đồ thị chuyển trạng thái với cơ sở dữ liệu, đó là hai bảng: VERTEX và TRANSITION\_STATE, có nội dung như trong hình 8, và bảng OBJECT\_POS (P\_ID, Vertex\_From, Vertex\_To, Times) để theo dõi tần suất  $t$ , trong thuộc tính Times.

• Kiểm tra RBTVTG

Khi dữ liệu của những đối tượng được nhập vào các bảng trong cơ sở dữ liệu bệnh viện, chúng được kiểm tra RBTVTG bằng những thủ tục như đã nói trong phần 5 và dữ liệu trong những bảng liên kết với đồ thị như đã nói ở phần đoạn liên trên.

TRANSITION STATE			VERTEX	
Curr state	Label	Trans state	V ID	Vname
untreated	l <sub>2</sub>	surgery	0	untreated
untreated	l <sub>3</sub>	radiation	1	surgery
untreated	l <sub>4</sub>	chemotherapy	2	radiation
surgery	l <sub>1</sub>	untreated	3	chemotherapy
surgery	l <sub>5</sub>	watching	4	watching
radiation	l <sub>1</sub>	untreated	5	recovered
radiation	l <sub>5</sub>	watching		
chemotherapy	l <sub>1</sub>	untreated		
chemotherapy	l <sub>5</sub>	watching		
watching	l <sub>6</sub>	recovered		

Hình 8. Những bảng liên kết với đồ thị chuyển trạng thái

## VII. KẾT LUẬN VÀ HƯỚNG NGHIÊN CỨU TIẾP THEO

Bài báo đã trình bày một phương pháp kiểm tra RBTVTG trên một cấu trúc toán rời rạc là đồ thị chuyển trạng thái và giải quyết khi

đồ thị có chu trình đi qua nhiều hơn một đỉnh, kết quả cho thấy, phương pháp đề xuất về kiểm tra RBTVTG khi nhập liệu có hiệu quả và dễ cài đặt, đồng thời nó trở thành cơ sở cho việc nghiên cứu tiếp tục, để giải quyết những yêu cầu về RBTVTG trong những ứng dụng thực tế, đó là: khi có nhiều phiên bản RBTVTG cùng tồn tại trong một cơ sở dữ liệu. Nội dung của những RBTVTG này cũng thay đổi, hay mất hiệu lực, hay RBTVTG mới được thêm vào. Nói cách khác, RBTVTG tiến hoá theo thời gian hay RBTVTG mang tính động.

## TÀI LIỆU THAM KHẢO

1. *J. Chomicki*; Efficient Checking of Temporal Integrity Constraints Using Bounded History Encoding; Technical Report KS66506, Computing and Information Sciences, Kansas State University Manhattan, 1995.
2. *J. Chomicki, D. Toman*; Implementing Temporal Integrity Constraints Using an Active DBMS; IEEE Transaction on Knowledge and Data Engineering, 7(4), 1995, pp. 566-582.
3. *J. Chomicki, D. Toman*; Temporal Logic in Information Systems; BRICS Lecture Series, Dept. of Computer Science, University of Aarhus, Denmark, Nov. 1997
4. *M. Gertz, U. W. Lipeck*; Temporal Integrity Constraints in Temporal Databases; Proceedings of the International Workshop on Temporal Databases, J. Clifford, A. Tuzhilin (Eds.), Sept. 1995, Workshops in Computing, Springer-Verlag, Berlin, pp. 77-92.
5. *M. Gertz, U. W. Lipeck*; Deriving Integrity Maintaining Triggers from Transition Graphs; Proceedings 9<sup>th</sup> International Conference on Data Engineering, IEEE Computer Society Press, 1993, pp.22-29.
6. *U.W. Lipeck, H. Zhou*; Monitoring Dynamic Integrity Constraints on Finite State Sequences and Existence Intervals; Proceedings of the 3<sup>rd</sup> International Workshop on Foundations of Models and Languages for Data and Objects, Aigen, Sept. 1991, pp. 115-130.
7. *Phạm Văn Chung*; Checking Temporal Integrity Constraints in Temporal Databases; Technical Report, Faculty of Information Technology, HCM City University of Technology, 2005.

Địa chỉ liên hệ: Nguyễn Thành Sơn - Tel: (08) 3743.0855, Mobi: 0918.648.899  
 Khoa Công nghệ Thông tin, Trường Đại học SPKT Tp. HCM  
 Số 1, Võ Văn Ngân, Quận Thủ Đức, Tp Hồ Chí Minh.