Tạp Chí Tin Học và Điều Khiển Học
Tập X(1994), Số 2, 29-35

Về các phép mã hoá hợp lý

Huỳnh Hữu Nghĩa

I. Một số định nghĩa

1. Quan hệ tổng quát **R** gồm tập các thuộc tính dR^+ và tập các ràng buộc phụ thuộc F. Ta viết $R =< R^+, F >$.
2. Xem hai tập $A, B \subseteq R^+$. Nếu B phụ thuộc hàm vào A ta ký hiệu $A \to B \in F^{**}$, với F^{**} là bao đóng của F suy từ hệ luật dẫn Amstrong.
3. Cho hai tập $A, B \subseteq R^+$: ta ghi $A \vee B$ là hội của hai tập A và B, $A \wedge B$ là giao của hai tập này.
4. Xem $A \subseteq R^+$. Tập hợp:

$$\mathrm{Cl}_F(A) = \{b \in R^+ : \exists A' \in A : A' \to b \in F^{**}\}$$

được gọi là bao đóng của tập A ứng với tập fth (phụ thuộc hàm) F.
5. Cho $A, B \in R^+$ và $A \to B$ F^{**}. Ta nói $A \to B$ là một fth nguyên tố nếu và chỉ nếu: mọi $A' \subset A$, nếu $A \to B \in F^{**}$ thì $A = A'$.
6. Xem tập $K \subset R^+$. K được gọi là chìa khóa của quan hệ $R =< R^+, F >$ nếu $K - R^+$ nguyên tố trong F^{**}.
7. Một fth $A - B \in F^{**}$ được gọi là fth nội bộ của một quan hệ $< T_i^+, F_i >$ nếu $A \vee B \subseteq T_i$ và A là một chìa khóa của $< T_i^+. F_i >$.

2. Cách mã hóa hợp lý trên tập cách biệt các thuộc tính

Xem quan hệ tổng quát $R =< R^+, F >$. Ta gọi một phân hoạch (partition) $M(R^+)$ trên tập R^+ là một cách mã hóa hợp lý trên quan hệ R nếu $M(R^+)$ thỏa mãn các điều kiện sau:

1. Từng vế trái và từng vế phải của từng fth thuộc F là hội cách biệt của một số tập trong $M(R^+)$.
2. Tập các thuộc tính không thuộc về phía nào của F là hội cách biệt của một số tập trong $M(R+)$.
3. Hai tập T_i, $T_j \in M(F)$ nếu $T_i \wedge T_j \neq 0$ thì $T_i = T_j$.

Ta dự định gán cho mỗi tập trong $M(R^+)$ một con số từ 1 đến m (duy nhất). Như vậy ta biến đổi tập F thành tập $M(F)$: tập "các fth với các vế gồm các con số".

Nếu một tập hợp $A \subseteq R^+$, A sẽ được gọi là một tập mã hóa được theo phép M. Ta ghi $M(A)$ là tập các mã số của các tập con thành phầcủa A.

III. Một số tính chất của một phép mã hóa hợp lý M

Ta xem một cách mã hóa hợp lý $M(R^+) = \{T_1, ..., T_n\}$.

(III.1) Mệnh đề:

Với $0 \leq m \leq n$, đặt T^m là hội tụ của m tập hợp nào đó thược $M(R^+)$. Họ $\{T_m\}$ sẽ đóng kín với các phép tính giao, hội, hiệu các tập hợp.

(III.2) Mệnh đề:

A và B là hai tập thược tính mã hóa được (theo M), $A = B$ nếu và chỉ nếu $M(A) = M(B)$.

(III.3) Định lý:

Cho $<M(R^+), M(F)>$ là kết quả mã hóa của một phép mã hóa M áp dụng trên quan hệ $<R+, F>$. Xem $A \to \{x\}$ là một fth không tầm thường nào đó. Thế thì

(1) Tồn tại một tập hợp $A_0 \subseteq A$, A_0 mã hóa được, $A_0 \to \{x\}$ F^{**}. Ta biểu diễn n *được bằng sơ đồ:*

$$\begin{tikzcd} A \arrow[r] \arrow[d] & A_0 \arrow[dl, dash] \\ x & \end{tikzcd}$$

Ghi chú: Ta nói $A \to x$ là kết quả bắc cầu giữa $A \to A_0$ (tầm thường) và $A_0 \to x$ với A_0 mã hóa được.

(2) tồn tại một tập $B \subseteq M(R^+)$ sao cho

- A và B cách biệt

- tồn tại một tập $A_0 \subseteq A$, A_0 mã hóa được và A_0 xác định hàm B. Ta biểu diễn bằng đồ thị:

$$\begin{tikzcd} A \arrow[r] \arrow[d] & A_0 \arrow[d] \\ x & B \arrow[l] \end{tikzcd}$$

(Ta nói rằng một fth bất kỳ $A \to x$ không tầm thường có thể được xấp xỉ bằng một fth $A_0 \to B$ với A_0 và B là hai tập mã hóa được).

(III.4) Định lý:

Xem tập $K \subseteq R^+$. T là một tập thược $M(R^+)$.
Xem $T_1 \subset T$. Ta có

$$Cl[(KDT) \vee T_1] = Cl(K \setminus T) \vee T_1$$

(ta thấy các tập "lỡ cỡ" không có giá trị gì trong phép tính bao đóng)

(III.4.1) Hệ quả

Xem tập $K \subseteq R^+$. Gọi K_1 là hội tất cả các tập con mã hóa được. chứa trong K. Đặt $K_2 = K \setminus K_1$. ta có:

$$Cl_F(K) = CL_F(K_1) \vee K_2.$$

(III.5) Mệnh đề

K là một tập mã hóa được thì $Cl_F(K)$ là một tập mã hóa được.

(III.6) Mệnh đề

A là một tập mã hóa được.-Thế thì

$$Cl_{M(F)}M(A) = M[Cl_F(A)]$$

(một phép mã hóa hợp lý không làm thay đổi kết quả của phép tính bao đóng).

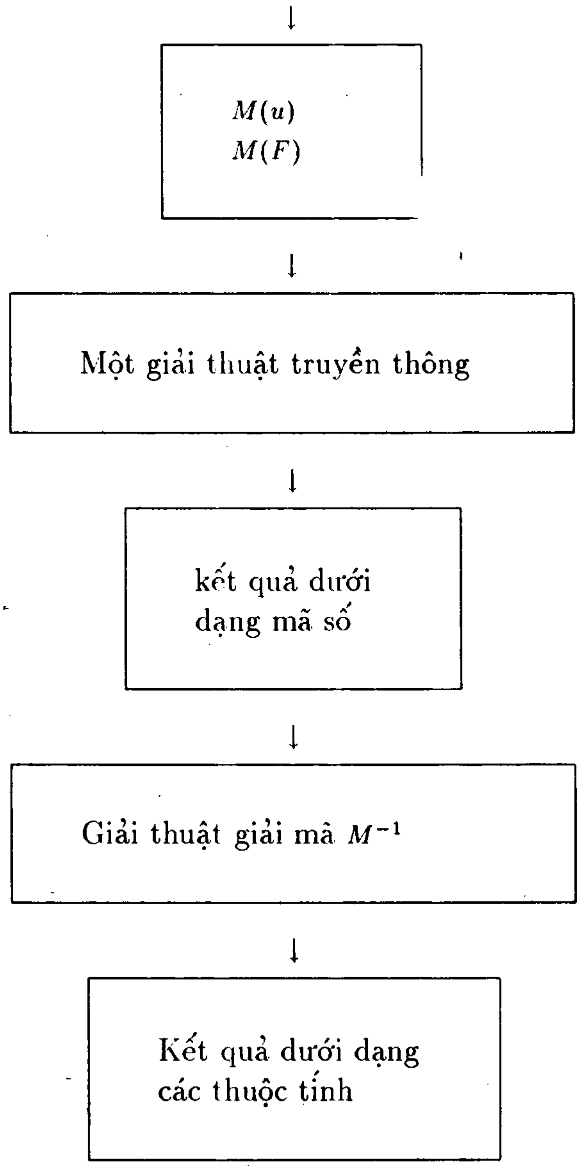
IV. Thực hiện các giải thuật truyền thống có dùng thêm cách mã hóa hợp lý

Chúng ta thử xem một tiến trình sau:

u, F

↓

Một giải thuật mã hoá hợp lý M



Bằng các kết quả toán học trong mục (III) chúng tôi chứng minh được rằng kết quả cuối cùng không thay đổi so với cách làm thông thường là gán cho mỗi thuộc tính một mã số, khi giải các bài toán truyền thống nói trên. Nói cụ thể hơn chúng tôi đã áp dụng có kết quả các giải thuật

- Tìm tất cả các chìa khóa của R, giải thuật Lucchessi-Osborn (1978) [5].
- Giải thuật tìm một cơ sở tối thiểu [4]
- Giải thuật quan niệm lược đồ CSDL nhất quán và đầy đủ [1]

Hơn nữa khi tiến hành nghiên cứu áp dụng giải thuật [1] chúng tôi chứng minh được rằng việc biểu diễn các fth "có nguy cơ gây mâu thuẫn dữ liệu" sẽ được thực hiện dễ dàng và trọn vẹn nhờ các vế của chúng đều là các tập mã hóa được.

V. Giải thuật mã hóa hợp lý tạo ra ít mã số nhất.

Chúng tôi xin giới thiệu một giải thuật đã được chứng minh là một phép mã hóa hợp lý và tạo ra ít mã số nhất (xem mục lục D.II.1 và D.II.2 trong [3]).

Trong giai đoạn mã hóa, chúng ta cần đến số thuộc tính, số phụ thuộc hàm được nhập vào và lập các sơ yếu lí lịch của chúng bằng các đặc tả

1) var tsốtht = integer; tsốth = integer;

Để dễ theo dõi, các cấu trúc lưu trữ chuẩn bị lưu trữ tối đa 24 thuộc tính và 100 phụ thuộc hàm

```
2) type   con_trỏ_tht = ^tht;
3) var    tht = record { ứng từng bước thuộc tính }
            tên = char;
            vị trí = array [1..200] of integer;
            { thứ tự của vế, trong cấu trúc FTH, có chứa thuộc tính này }
4) var    quan_hệ = array [1..24] of con_trỏ_tht
5) type     vế = set of [A..Z];
            tập_con = set of [1..24];
6) var tập_fth = array[1..200] of vế;
            mã_fth = array[1..200] of tập_con;
```

Cấu trúc tập_fth chừa 100 thành tố đầu để chứa toàn các vế trái và 100 thành tố sau để chứa toàn các vế phải. Cấu trúc mã_fth để chứa kết quả mã hóa của tập phụ thuộc hàm.

```
7) var   CHKHOA = array [1..24] of vế
```

Cấu trúc CHKHOA dùng để giải mã một tập hợp gồm các mã số thành một tập các thuộc tính.

Tóm lược giải thuật mã hóa NGH

(bước 1) *Sơ chế dữ liệu nhập*

(1.1) - Sát nhập các fth có cùng vế trái lại để tạo ra một fth mới có vế phải bằng hội các vế phải cũ.

- Loại bỏ các thuộc tính ở vế phải của một fth, đã có mặt ở vế trái. Ví dụ: biến đổi $ABC \to ADC$ thành $ABC \to D$

(1.2) - Đếm số fth và số thuộc tính nhập vào

- Ghi lại thuộc tính nào ở vế nào của fth nào và nhờ vậy biết được có tất cả bao nhiêu vế chứa thuộc tính đó.

(bước 2) *Sắp xếp array các con trỏ quan_hệ để chuẩn bị xử lý các thuộc tính theo chiều giảm dần tổng số vế có chứa thuộc tính đó*

(bước 2) *Lặp đi lặp lại đến hết các thuộc tính thuộc ít nhất một vế.*

(3.1) Xem thuộc tính x.

(3.2) Xét từng thuộc tính y có cùng tổng số vế với x.

Nếu x và y có mặt trong các vế:

(a) Đưa x và y vào chung một tập CHKHOA [i] nào đó và tập này mang mã số là i

(b) Che thuộc tính y lại để lần sau không xét nữa.

(c) Ghi mã số i vào cấu trúc của mã_fth. Cụ thể là nếu x thuộc các vế $v_1, ..., v_m$ thì mã số i được ghi vào các thành tố thứ $v_1, ..., v_m$ của array mã_fth.

(bước 4) *Gom tất cả thuộc tính không thuộc vế nào vào tập chung CHKHOA[24].*

(bước 5) *Trả về các trị mã fth*□ *và CHKHOA*□

Nhận xét

(1) Ở bước 3 ta thao tác các thuộc tính theo các tuần tự trên một hàng đợi:

var QUEUE = array[1..24] of boolean

- chỉ xử lý những thành tố có giá trị 1

- những thuộc tính đã giải quyết xong (dà bị che) thì thành tố tương ứng sẽ có giá trị 0

(2) Độ phức tạp của các giải thuật trên là

$$|U|^2 + |U|.|F|.$$

- $|U|$ là các số thuộc tính được nhập vào

- $|F|$ là độ dài lưu trữ các fth

- $|U|^2$ là độ dài phức tạp (tối đa) của một thuật toán sắp thứ tự (xấu nhất) được xử dụng đến.

Chúng tôi chân thành biết ơn TS Trần Hà Nam, Công ty SCITEC thành phố HCM đà bảo trợ tác giả hoàn thành bài báo này.

Các tài liệu tham khảo

1. Bích Thủy, Lương Đồng Thi, *Une approache de conception: une base de données cohérentes et complete.* Luận văn tiến sĩ số 314 khoa KH KTXH trường đại học Gènève, 1986.
2. Ultman J.D., *Principles of database systems,* 2^{nd} edition, 1982.
3. Huỳnh Hữu Nghĩa, *Các phép mã hóa hợp lý,* tiểu luận chưa công bố, 1989.
4. Beeri C. & Berstein P.A., *Computational problems related to the design of normal forms for relational schemas,* ACM Trans. on DB systems, v.4, n. 1, 1979.
5. Hồ Thuần & Lê Văn Bào, *Sufficient conditions for which a Relation Scheme has precisely one key,* Preprint Series n. 7, 1983, Viện KHTT và ĐK Việt Nam.

Abstract

On logical codings in process designing a relational database scheme with functional dependencies.

In process to implement a database scheme with "good" qualities, at the first step, we often assign to each of the set of entities a number (a code). Making note that almost the complexity of conceptual algorithms is usually, at least, proportional to number of codes of attributes, and to the length of the dependency set, we study methods numerating subsets of a "logical" partition of the set of attributes. Such methods are so-called "logical Codings".

In this paper, we study Logical Codings on an universal $U = < S, F >$, *in which* S *is the set of attributes,* F *is the set of functional dependencies. Those Logical Codings are to be proven independent to all conceptual algorithms.*

And at the final, we introduce the Logical Coding NGH which has low complexity and produces a minimal set of codes.