TẠP CHÍ KHOA HỌC TÍNH TOÁN VÀ ĐIỀU KHIỂN TẬP VI (1990), Số 4 (6-15)

CÁC THUẬT TOÁN QUI HOẠCH ĐỘNG SONG SONG ĐỂ NHẬN DẠNG ẢNH DỰA TRÊN ĐƯỜNG BIÊN

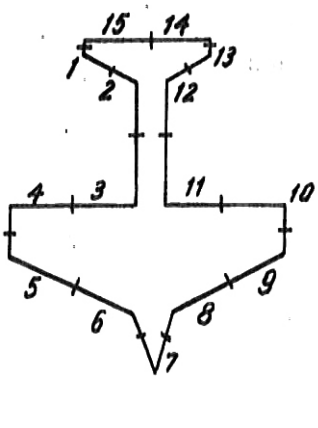
GIANG VŨ THẮNG
Viện Tin học, Viện KHVN

I - GIỚI THIỆU

Trong bài báo này, chúng tôi trình bày phương pháp song song hóa thủ tục qui hoạch động cho một số mô hình tính toán song song khác nhau. Trong phần II của bài sẽ trình bày tóm tắt việc ứng dụng của qui hoạch động trong vấn đề đối sánh ảnh dựa trên đường chu tuyến của nó. Phần III mô tả thuật toán song song của thủ tục này cho mô hình mảng systolic 2 chiều với k^2 bộ xử lý có sẵn. Ở đây $k = \max(2n, m)$, m là số các đoạn của chu tuyến cần được nhận dạng và n là số các đoạn của chu tuyến mẫu. Độ phức tạp tính toán của thuật toán là $O(k)$. Tiếp đó, phần IV sẽ trình bày một phương án cải tiến của thuật toán này cho mảng systolic một chiều với số bộ xử lý cần thiết ít hơn rất nhiều nhưng vẫn đạt cùng mức độ phức tạp tính toán. Trong phần V, việc ước lược giá trị của con đường có độ đo khoảng cách cực tiểu trong một bảng trên máy tính associative kiểu SIMD với $2k - 1$ bộ xử lý sẽ được giải quyết. Một thuật toán song song associative được xây dựng với độ phức tạp tính toán là $O(kw)$, ở đây w là độ rộng tính toán theo bít của các trường associative đã được xử dụng.

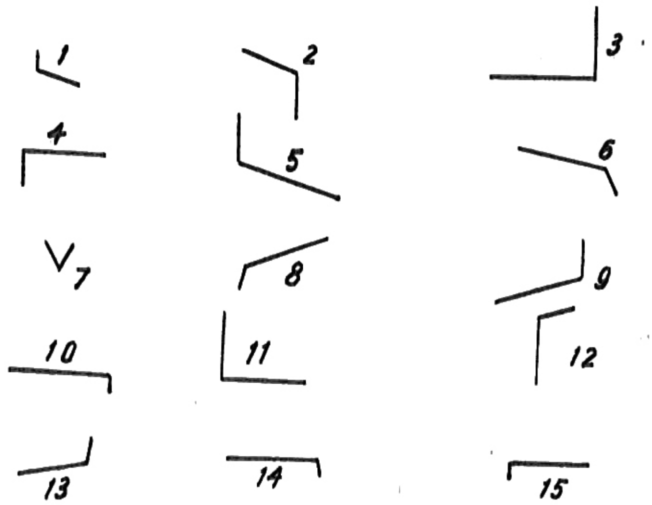
II - ỨNG DỤNG CỦA KỸ THUẬT QUI HOẠCH ĐỘNG TRONG ĐỐI SÁNH DẠNG DỰA TRÊN CHU TUYẾN CỦA NÓ

Trong [1], một ứng dụng của kỹ thuật qui hoạch động trong đối sánh dạng dựa vào chu tuyến đã được trình bày một cách chi tiết. Tư tưởng chính của phương pháp này là tách chu tuyến mẫu ra thành một số đoạn cơ sở và việc đối sánh giữa chu tuyến mẫu và chu tuyến của đối tượng cần nhận dạng được thực hiện thông qua các tham số đặc trưng của các đoạn cơ sở này.



Hình 1

Hình 1 cho chu tuyến của một mẫu máy bay, Chu tuyến này đã được xác định và được biểu diễn bởi mã xích. Sau đó chu tuyến được phân chia thành một số đoạn cơ sở phân biệt được đánh số từ 1 đến n (ví dụ $n = 15$, xem hình 2).



Hình 2

Bằng việc sử dụng các hệ số Fourier tính được từ các đoạn, chúng ta có thể mô tả các đoạn cơ sở này một cách hiệu quả bởi 6 hệ số Fourier như sau

$$F^i = (F_1^i, F_2^i, F_3^i, F_4^i, F_5^i, F_6^i, F_{-6}^i, F_{-5}^i, F_{-4}^i, F_{-3}^i, F_{-2}^i, F_{-1}^i)$$

$i = \overline{1, n}$.

F_j^i là hệ số Fourier thứ j của đoạn trích được thứ i. Bằng một kỹ thuật chuẩn hóa đã được trình bày chi tiết trong [1], chúng ta sẽ đạt được véc tơ chuẩn hóa SF^i của véc tơ F^i trong dạng sau

$$SF^i = (SF_1^i, SF_2^i, SF_3^i, SF_4^i, SF_5^i, SF_6^i, SF_{-6}^i, SF_{-5}^i, SF_{-4}^i, SF_{-3}^i, SF_{-2}^i, SF_{-1}^i)$$

Chúng tôi không trình bày ở đây chi tiết thủ tục chuẩn hóa các hệ số Fourier. Độc giả có thể tham khảo trong [1] để có được các thông tin chi tiết.

Để thực hiện việc nhận dạng một mẫu mới với m véc tơ của hệ số Fourier đã được chuẩn hóa $SA^j = (SA_1^j, SA_2^j, SA_3^j, SA_4^j, SA_5^j, SA_6^j, SA_{-6}^j, SA_{-5}^j, SA_{-4}^j, SA_{-3}^j, SA_{-2}^j, SA_{-1}^j)$ ở đây $j = \overline{1, m}$. Một hàm độ đo khoảng cách giữa hai chu tuyến được tính bởi công thức sau

$$D(SF^i, SA^j) = a_{ij} = \sum_{k=-6}^{6} |SF_k^i - SA_k^j|^2, \ i = \overline{1, n}, \ j = \overline{1, m}$$

Dựa vào hàm khoảng cách này, một bảng độ đo khoảng cách giữa hai chu tuyến sẽ được thiết lập như sau

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}$$

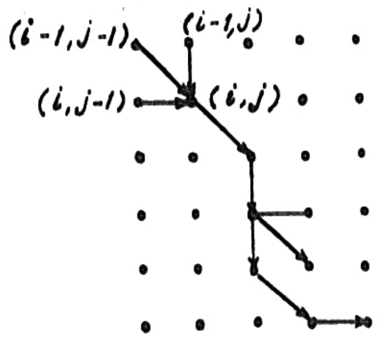
Ở đây m là số đoạn của chu tuyến cần nhận dạng còn n là số đoạn của chu tuyến mẫu.

Khoảng cách giữa hai chu tuyến được định nghĩa là giá trị đo được của con đường có độ đo khoảng cách cực tiểu đi ngang qua bảng khoảng cách A. Con đường này xuất phát tại một vị trí nào đó trong cột đầu tiên của bảng A và kết thúc tại một điểm nào đó trong cột cuối của bảng này (xem hình 3 để có sự giải thích chi tiết hơn).

Con đường có độ đo khoảng cách cực tiểu trong bảng A được tìm ra nhờ việc áp dụng kỹ thuật qui hoạch động. Kỹ thuật này hoạt động dựa theo một nguyên tắc quan trọng sau: "con đường có độ đo khoảng cách cực tiểu xuất phát từ cột đầu tiên của bảng A và kết thúc tại cột cuối cùng là việc hợp nhất của con đường có độ đo khoảng cách cực tiểu xuất phát từ cột đầu tiên tới một điểm a_{ij} nào đó của bảng và con đường từ điểm đó tới cột cuối của bảng". Chiến lược tìm kiếm con đường có độ đo khoảng cách cực tiểu được biểu diễn bởi công thức sau

$$D(i,j) = \min \begin{cases} a_{i+1,j+1} & + \quad D(i+1, j+1) \\ a_{i+1,j} & + \quad D(i+1, j) \\ a_{i,j+1} & + \quad D(i, j+1) \end{cases} \tag{1}$$

Ở đây $D(i,j)$ là trị số của con đường có độ đo khoảng cách cực tiểu xuất phát tại điểm a_{ij} và kết thúc tại cột cuối cùng của bảng A.



Hình 3

Do điểm bắt đầu của con đường mong muốn là còn chưa biết nên cần thiết là phải xem xét tất cả các khả năng có thể và lặp lại hai lần các dòng của bảng A (xem hình 3a).

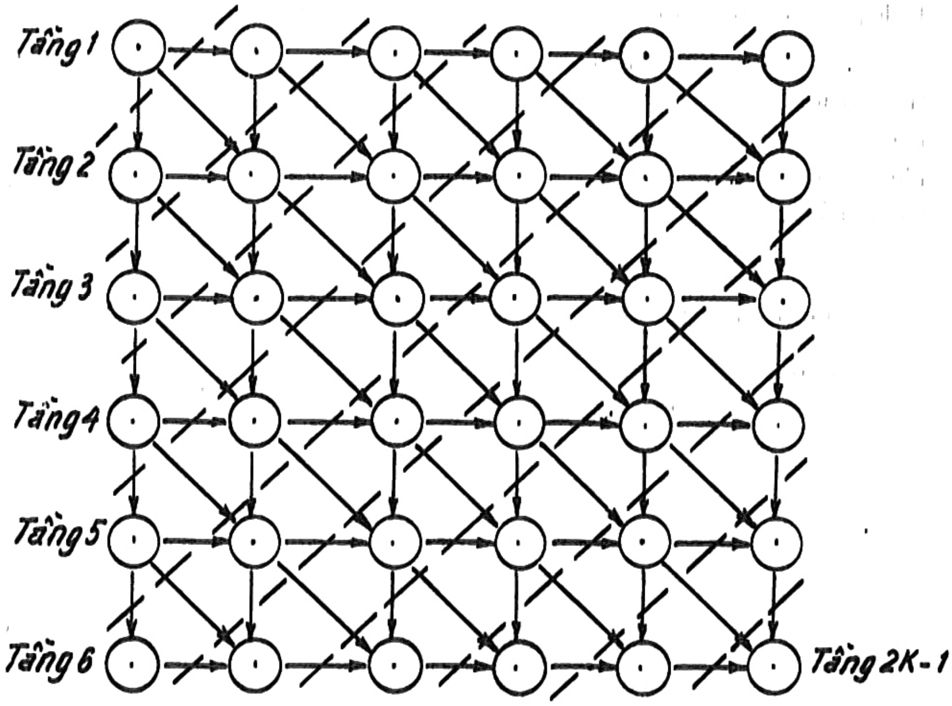
$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & & & \\ a_{n1} & a_{n2} & \dots & a_{nm} \\ a_{11} & a_{12} & \dots & a_{1m} \\ \dots & & & \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix}$$

Hình 3a

Một trong các dạng có thể có của con đường mong muốn được chỉ ra trong hình 3, tương ứng với chiến lược tìm kiếm (1), $m = 5$ và $2n = 6$.

Một đối tượng mới được nhận biết thuộc vào một dạng mẫu nào đó nếu trên dạng mẫu này con đường có độ đo khoảng cách cực tiểu là giá trị nhỏ nhất trong các con đường có độ đo khoảng cách cực tiểu ứng với các dạng mẫu khác.

III – THUẬT TOÁN TÌM KIẾM SONG SONG TRÊN MẢNG SYSTOLIC 2 CHIỀU



Hình 4

Một trong những lý do chính để giải bài toán trên hệ thống mảng systolic là do tính hiệu quả của việc truyền dẫn thông tin giữa các bộ xử lý của mảng.

Giá trị cũ tại một điểm của bảng khoảng cách sẽ được cập nhật phụ thuộc vào các giá trị tại 3 điểm lân cận khác của điểm này (đó là lân cận bắc, lân cận tây bắc và lân cận phía tây). Sau đó các giá trị mới của các điểm đã được xem xét sẽ được truyền tới 3 điểm lân cận của chúng theo các hướng đông, đông nam và hướng nam (xem hình 3).

Cấu hình của mảng systolic được xây dựng như sau: Nó là mảng systolic 2 chiều có kích thước $k \times k$, ở đây $k = \max(2n, m)$ (xem hình 4).

Mảng systolic bao gồm k^2 bộ xử lý, chúng được ký hiệu bởi P_{ij}, $i = \overline{1,k}$, $j = \overline{1,k}$. Mọi bộ xử lý P_{ij} bao gồm 3 thanh ghi R_1, R_2, R_3, một vùng nhớ riêng cục bộ, ở đó giá trị a_{ij} $i, j = \overline{1,k}$ được lưu giữ và một thanh ghi trạng thái R_S. Các bộ xử lý P_{ij} với các chỉ số i, j không tồn tại trong bảng khoảng cách A sẽ lưu giữ giá trị 0 trong vùng nhớ riêng cục bộ của nó. Ba thanh ghi R_1, R_2, R_3 sẽ nhận các thông tin được truyền tới từ các bộ xử lý lân cận theo các hướng tây, tây bắc, bắc.

Các bộ xử lý P_{ij} $(i, j = \overline{1,k})$ được gọi là đang kích hoạt ($R_S = 1$) khi và chỉ khi nội dung của cả 3 thanh ghi R_1, R_2, R_3 đã được nạp giá trị. Đầu tiên tất cả các bộ xử lý P_{i1} $i = \overline{1,k}$ của cột đầu tiên được kích hoạt, các thanh ghi R_1, R_2, R_3 của chúng được nạp các giá trị 0. Các bộ xử lý P_{1j}, $j = \overline{2,k}$ của hàng đầu tiên sẽ ở trạng thái không hoạt động, các thanh ghi R_2, R_3 được nạp giá trị G (G là giá trị lớn nhất mà các thanh ghi có thể lưu giữ). Một bộ xử lý chỉ có thể nạp các giá trị được truyền tới từ các bộ xử lý lân cận vào các thanh ghi R_1, R_2, R_3 của nó khi và chỉ khi nó chưa được kích hoạt. Tất cả các bộ xử lý được kích hoạt của mảng systolic sẽ thực hiện cùng một phép toán tại mỗi nhịp tính toán, chúng bao gồm 2 phép so sánh, 1 phép cộng và 1 phép truyền dữ liệu tới ba bộ xử lý lân cận của mình.

Các bộ xử lý của mảng systolic làm việc một cách đồng bộ tại mỗi một nhịp thời gian, tuy nhiên thông tin lan truyền được diễn ra một cách tuần tự theo hướng đường chéo chính của mảng. Sau mỗi bước tính toán, một nhóm bộ xử lý nằm trên đường song song với đường chéo phụ của mảng sẽ được kích hoạt. Các nhóm bộ xử lý này sẽ hình thành $(2k-1)$ tầng khác nhau trong mảng systolic (xin xem hình 4 để có sự giải thích chi tiết).

Thuật toán song song cho mảng systolic 2 chiều có thể được viết hình thức trong dạng sau:

Thuật toán 1

Trong khi P_{kk} chưa được kích hoạt

Do {Với tất cả các bộ xử lý đã được kích hoạt P_{ij}}

Begin

$TG \longleftarrow \min(R_1, R_2, R_3)$

$a_{ij} \longleftarrow a_{ij} + TG$

Truyền giá trị của a_{ij} theo hướng đoong,

đông nam và nam.

End

Ta dễ dàng thấy rằng thuật toán 2 sẽ kết thúc sau $2k-1$ phép lặp và cho ta các con đường với độ đo khoảng cách cực tiểu xuất phát từ cột đầu tiên và kết thúc tại cột cuối cùng của bảng A, giá trị của chúng được lưu giữ trong bộ nhớ của các bộ xử lý P_{ik} $i = \overline{1,k}$.

Để chọn giá trị nhỏ nhất trong các giá trị này, $k-1$ phép so sánh là cần thiết với việc áp dụng một thuật toán tuần tự thông thường.

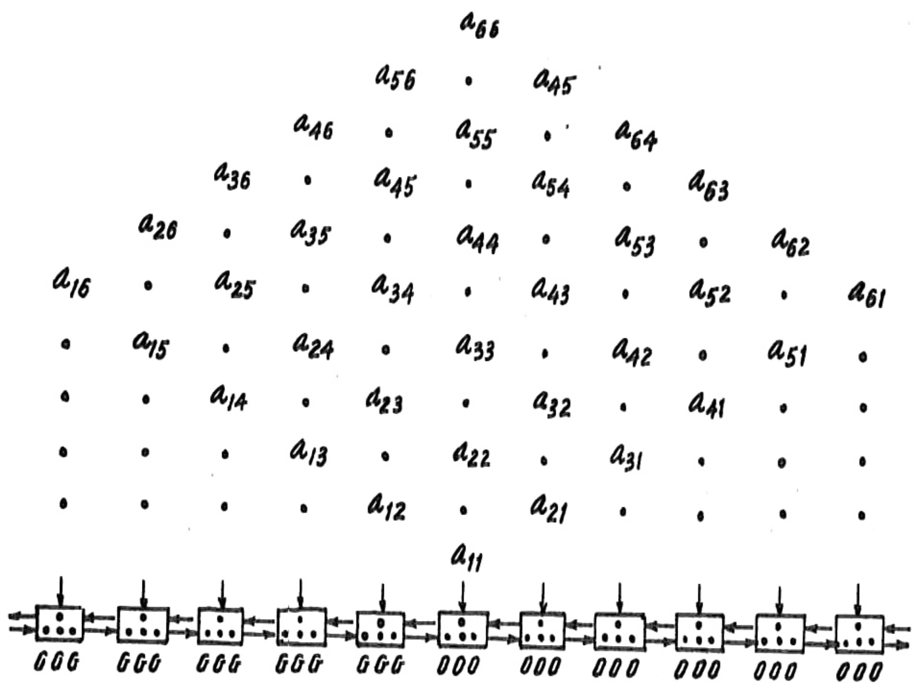
Tóm lại, độ phức tạp tính toán tổng cộng của thuật toán tìm kiếm con đường có độ đo khoảng cách cực tiểu trên mảng systolic 2 chiều với k^2 bộ xử lý có sẵn là $O(k)$, ở đây $k = \max(2n, m)$.

IV - THUẬT TOÁN SONG SONG CẢI TIẾN CHO MẢNG SYSTOLIC 1 CHIỀU

Bài toán ước lược giá trị của con đường có độ đo khoảng cách cực tiểu trình bày trong phần III còn một số nhược điểm mà chúng ta cần phải vượt qua. Số các bộ xử lý cần xử dụng là quá lớn, có nghĩa là mỗi bộ xử lý chỉ lưu giữ một số hạng trong bảng A. Nhược điểm nữa là có quá nhiều bộ xử lý nhàn rỗi trong quá trình thực hiện thuật toán. Trong mỗi bước tính toán, chỉ một nhóm các bộ xử lý nằm trên đường song song với đường chéo phụ của mảng là được kích hoạt và làm việc. Đó là một nhược điểm lớn mà chúng ta cần tránh.

Phần này sẽ trình bày phương án cải tiến của thuật toán nói ở trên. Một mảng systolic 1 chiều với $2k-1$ bộ xử lý $(P_1, P_2, \ldots, P_{2k-1})$ sẽ được xây dựng. Mỗi bộ xử lý P_i $(i = 1, 2, \ldots, 2k-1)$ có 3 thanh ghi, được ký hiệu là R_r^i, R_w^i, R_l^i và một bộ nhớ nhỏ M^i để lưu giữ dữ liệu được nạp từ bảng A. Thanh ghi R_r^i nhận dữ liệu được truyền từ bộ xử lý R_{i-1}, thanh ghi P_l^i nhận dữ liệu được truyền từ P_{i+1}. Thanh ghi R_w^i lưu cất giá trị đã được xử lý của P_i cho bước tính toán tiếp sau của nó.

Một bộ xử lý được gọi là kích hoạt khi và chỉ khi bộ nhớ M của nó đã nạp dữ liệu từ bảng A. Mỗi bộ xử lý P_i có 2 lối ra để truyền đi các dữ liệu của nó đã được xử lý và 3 lối vào để nạp dữ liệu vào từ bảng A và từ hai bộ xử lý lân cận. Cấu hình của mảng systolic 1 chiều với việc bố trí dữ liệu và sự phân chia thời gian cho việc nạp dữ liệu có thể được thấy trong hình 4a với $k = 6$.



Hình 4a

Trước hết, nội dung của các thanh ghi R_r^i, R_w^i, R_l^i của $k-1$ bộ xử lý đầu tiên P_i, $i = 1, 2, \ldots, k-1$ được gán bằng G (G là số lớn nhất mà thanh ghi R có thể lưu giữ). Trong khi các thanh ghi R_r^i, R_w^i, R_l^i của k bộ xử lý còn lại $P_i (i = k, k+1, \ldots, 2k-1)$ được khởi động bởi 0. Hoạt động của bộ xử lý P_i có thể được mô tả theo cách sau. Đầu tiên dữ liệu của bảng A được nạp vào bộ nhớ cục bộ M^i của bộ xử lý P_i. Nội dung của 3 thanh ghi R_r^i, R_w^i, R_l^i được so sánh với nhau để chọn giá trị nhỏ nhất trong chúng. Tiếp đó nội dung của M^i được cộng với giá trị nhỏ nhất này, tổng sẽ được lưu giữ trong R_w, Đồng thời tổng này cũng được truyền tới hai bộ xử lý lân cận P_{i-1} và P_{i+1} theo các hướng trái và phải.

Thuật toán bao gồm $2k-1$ phép lặp. Trong k bước lặp đầu tiên, bước tính toán thứ I ($I = 1, ..., k$ có I bộ xử lý được kích hoạt với các chỉ số của chúng lần lượt là $k - I + 2j + 1$, $j = 0, ..., i-1$. Trong $k-1$ bước lặp còn lại, bước tính toán thứ I ($I = k+1, ..., 2k-1$) có $2k - I$ bộ xử lý được kích hoạt, các chỉ số lần lượt là $(I-k) + 2j + 1$, $j = 0, ..., 2k - I - 1$.

Thuật toán có thể được viết trong dạng sau

Thuật toán 2

For $I = 1, 2, ..., 2k-1$

Nạp dữ liệu vào bảng từ bảng A

Do {Với tất cả các bộ xử lý đã được kích hoạt của mảng}

Begin

$TG^i \longleftarrow \min(|R_r^i|, |R_w^i|, R_l^i|)$

$M^i \longleftarrow M^i + TG^i$

(truyền dữ liệu đã được xử lý)

$R_w^i \longleftarrow M^i$

$R_l^{i-1} \longleftarrow M^i$

$R_r^i \longleftarrow M^i$

End

Thuật toán sẽ kết thúc sau $2k-1$ phép lặp và cho chúng ta giá trị nhỏ nhất của các con đường có độ đo khoảng cách cực tiểu ở k thanh ghi đầu tiên R_w^i ($i = 1, ..., k$) của mảng systolic.

Giá trị của con đường có độ đo khoảng cách cực tiểu được tìm thấy bởi thuật toán tìm giá trị cực tiểu song song được trình bày chi tiết trong [4] với độ phức tạp tính toán của nó là $O(k)$.

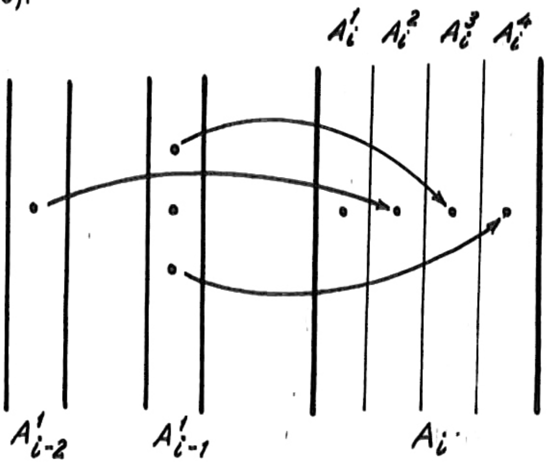
Tóm lại, độ phức tạp tính toán giành cho việc đánh giá trị số của con đường có độ đo khoảng cách cực tiểu trong bảng A đạt mức $O(k)$, trên $2k-1$ bộ xử lý của mảng systolic một chiều.

V – THUẬT TOÁN TÌM KIẾM SONG SONG TRÊN MÁY TÍNH ASSOCIATIVE KIỂU SIMD

Trong phần này, chúng tôi trình bày một thuật toán tìm kiếm kiểu qui hoạch động định hướng trên các máy tính song song associative kiểu SIMD.

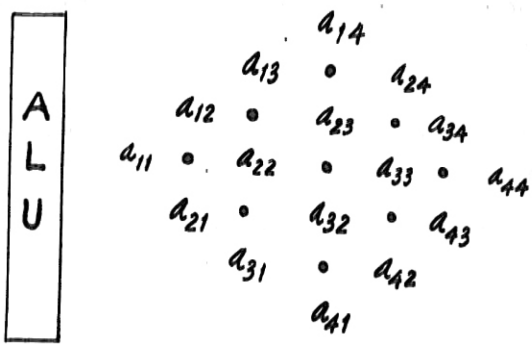
Đầu tiên chúng ta giả thiết rằng: độ rộng trường w bit là đủ để biểu diễn các độ đo khoảng cách giữa các đoạn hoặc giá trị của mỗi con đường có thể có trong bảng A.

Thuật toán sử dụng $2k = 1$ trường associative, được ký hiệu lần lượt là $A_1, A_2, ..., A_{2k-1}$. Mỗi trường A_i ($i = 1, ..., 2k-1$) được chia nhỏ thành 4 trường con ký hiệu bởi $A_i^1, A_i^2, A_i^3, A_i^4$. A_i^1 lưu giữ các giá trị khoảng cách giữa các đoạn của bảng A, A_i^2 lưu giữ các giá trị được truyền tới từ các trường A_{i-2}^1 ở cùng một mức, A_i^3 lưu giữ các giá trị được truyền tới từ trường A_{i-1}^1 ở mức cao hơn một vị trí và A_i^4 cất giữ các giá trị được truyền tới từ trường A_{i-1}^1 ở mức thấp hơn một vị trí (xem hình 5).



Hình 5

Để thực hiện một cách hiệu quả các phép toán trên các trường associative, đầu tiên các trường A_i^j $i = \overline{1, 2k-1}$, $j = 2, 3, 4$ sẽ được gán giá trị 0 và việc phân bố các giá trị khoảng cách giữa các đoạn a_{ij} trên các trường của bộ nhớ MDA sẽ được cho như sau với $k = 4$



Hình 6

Dấu • ký hiệu các vị trí mà tại đó chúng ta không quan tâm tới giá trị của nó. Trong thuật toán associative dưới đây, thanh ghi mặt nạ M có vị trí vô cùng quan trọng. Ta gọi $mask(u, v)$ là một hàm mặt nạ, nó tạo ra trong thanh ghi M của module associative u giá trị 0 tại u vị trí đầu tiên của thanh ghi M và v giá trị 1 tại v vị trí tiếp theo, ví dụ M = mask(3,4) sẽ tạo ra M = (00011110...0).

Thuật toán tìm kiếm song song theo kiểu qui hoạch động trên máy tính song song associative với một module associative và $(2k-1)$ bộ xử lý được cho trong dạng sau đây.

Thuật toán 3

For $i = 1, 2, \ldots, k$

$M \longleftarrow \text{mask}(k-i, 1)$

$A_i^2 \longleftarrow G(M)$

$A_i^3 \longleftarrow G(M)$

Continue

For $i = 1, 2, \ldots, 2k - 1$

$M \longleftarrow \text{mask}(|k - i|, 2i - 1)$

$TG \longleftarrow \min(A_i^2, A_i^3, A_i^4)$

$A_i^1 \longleftarrow (A_i^1 + TG)(M)$

(propagation)

$A_{i+2}^2 \longleftarrow A_i^1(M)$

$A_{i+1}^2 \longleftarrow \text{SHIFT}^{+1}(A_{i+1}^3)$

$A_{i+1}^3 \longleftarrow A_i^1(M)$

$A_{i+1}^3 \longleftarrow \text{SHIFT}^{-1}(A_{i+1}^3)$

$A_{i+1}^4 \longleftarrow \text{SHIFT}^{-1}(A_{i+1}^4)$

$A_{i+1}^4 \longleftarrow A_i^1(M)$

$A_{i+1}^4 \longleftarrow \text{SHIFT}^{+1}(A_{i+1}^4)$

Continue

Bằng việc lấy tổng độ phức tạp tính toán của macro lệnh đã được sử dụng trong thuật toán này, chúng ta có thể dễ dàng đánh giá được độ phức tạp của thuật toán với cấp $O(kw)$, ở đây $k = \max(2n, m)$, w là số bit cần thiết đủ để biểu diễn giá trị của mỗi con đường tìm kiếm trong bảng A.

Khi thuật toán 3 kết thúc, các giá trị khoảng cách của các con đường có độ đo khoảng cách cực tiểu đi từ cột đầu tiên tới cột cuối cùng của bảng A được định vị trên k trường cuối cùng trong bộ nhớ MDA.

Thuật toán song song để tính giá trị nhỏ nhất của các con đường có độ đo khoảng cách cực tiểu có thể có được chỉ ra dưới đây.

Cuối cùng giá trị của con đường mong muốn sẽ được lưu giữ tại phần tử đầu tiên từ trên xuống của trường associative A_k^1.

$M \longleftarrow \text{mask}(k - 1, 1)$

For $i = 2k - 2, 2k - 3, \ldots, k$

$TG \longleftarrow A_{i+1}^1$

$TG \longleftarrow \text{SHIFT}^{+1}(TG)$

$M \longleftarrow \text{SHIFT}^{+1}(M)$

$A_i^1 \longleftarrow \min(A_i^1, TG)(M)$

Continue

Hiển nhiên độ phức tạp tính toán $O(kw)$ đạt được trong thuật toán này.

Tóm lại, độ phức tạp tính toán toàn bộ $O(kw)$ là đạt được trên máy tính song song associative kiểu SIMD với một module associative cho bài toán tìm kiếm kiểu qui hoạch động.

VI - KẾT LUẬN

Qui hoạch động là một kỹ thuật quan trọng trong lĩnh vực xử lý tiếng nói và nhận dạng ảnh. Trong bài này, việc ứng dụng kỹ thuật nói trên cho vấn đề nhận dạng đối tượng dựa vào các đoạn cơ sở được trích ra từ chu tuyến của nó đã được trình bày. Ba phương án của thuật toán tìm kiếm song song kiểu qui hoạch động đã được xây dựng trên các mô hình khác nhau của máy tính song song, như mảng systolic 2 chiều, mảng systolic 1 chiều và máy tính associative kiểu SIMD. Các độ phức tạp tính toán $O(k), O(k), O(kw)$ lần lượt đạt được trên các mô hình tính toán song song này.

Nhận ngày 5-8-1990

TÀI LIỆU THAM KHẢO

1. John W., Gorman O., Robert Mitchell and Frank P. Kuhl - Partial shape recognition using dynamic programming. IEEE Transactions on PAMI, Vol. 10, No. 2, March 1988.

2. Basile Louka and Maurice Tchuente - Dynamic programming on 2D systolic array Technical report.

3. Richter K. - Parallel Computer System SIMD. In proc. AIICSR Conference - I. Plander, ed. - North Holland, 1984, pp. 309 - 313.

4. Miklosko J. Vajtersic, Klette R., Vrto I. - Fast algorithm and their imolementation on specialized parallel computers. Vada - North Holland, Bratislava - Ansterdam, 1989.

5. Alberto Martelli - An application of heuristic search methods to edge and contour detection. Computer methods in image analisis. Edited by J. K. Aggarwal, Richard O. Duda, Azriel Rosenfeld, pp. 217 - 227.

6. M. Vajtersic, Giang Vu Thang - Threshold and histogram algorithms for a parallel associative computer. Computer and Artificial Intelligence, 5 (1986), No. 2, pp. 143 - 161.

ABSTRACT

The dynamic programming is an important procedure for speech understanding and image recognition. In this paper, the application of dynamic programming to recognize the shape basing on its contour is recalled, then the parallelism of this process is analyzed and exploited. Some parallel dynamic programming algorithms for evaluating the smallest value of the minimum intersegment distance path in a data table are designed in detail on several kinds of the specialized paralled computer, such as 2D Systolic Array, 1D Systolic Array and an associative computer of the SIMD type. Their computation complexities are respectively estimated as $O(k)$, $O(k)$ and $O(kw)$, while the cost of sequential version is $O(mn)$, where $k = \max(2n, m)$, m is the number of segments of a contour to be recognized, n gives the number of segments of the sample contour and w is the bit-width of the used associative fields.