

Chương 1

MỞ ĐẦU

1.1. TỔNG QUAN

Một công việc, một vấn đề nào đó hay một bài toán nào đó,... (như xây dựng một công trình nhà ở, đô thị, cầu cống, may mặc, giải một bài toán,...) nếu được phân chia thành các vấn đề nhỏ hơn, các vấn đề nhỏ này nếu có thể thực hiện đồng thời, thì ta nói vấn đề đó được thực hiện song song. Sau khi tất cả các vấn đề con được thực hiện xong, các kết quả của chúng sẽ được kết nối lại để có kết quả chung cho cả vấn đề cần giải quyết. Chẳng hạn việc đào móng công trình và chuẩn bị sắt làm móng có thể làm đồng thời. Sau khi cả hai phần việc này xong, chúng ta có thể đổ móng cho công trình. Hoặc khi cộng hai vectơ n phần tử, chúng ta có thể phân nhóm các phần tử và chia cho các bộ xử lý cùng thực hiện phép cộng,...

Cuốn sách bao gồm những phần rất cơ bản về tính toán song song, và được dành cho những ai muốn nghiên cứu về vấn đề này. Việc học tập và nghiên cứu được bắt đầu với phần tổng quan ngắn gọn bao gồm các khái niệm và các thuật ngữ liên quan tới tính toán song song. Các chủ đề này cũng sẽ được thảo luận bởi các ấn phẩm liên quan đến lập trình song song. Cuốn sách cũng bao gồm một số ví dụ về cách song song hóa các chương trình tuần tự.

Tính toán song song là sự phát triển của tính toán tuần tự, tính toán song song tham gia cạnh tranh những cái đang tồn tại trong thế giới tự nhiên: rất phức tạp, các sự kiện liên quan với nhau xảy ra tại cùng thời điểm, cùng trong phạm vi một chuỗi hành động, chẳng hạn:

- Hệ hô hấp và hệ tuần hoàn của các loài động vật
- Việc hình thành dải Ngân Hà
- Vận động của hành tinh
- Đại dương và thời tiết
- Trôi dạt, tích tụ và kiến tạo địa tầng.
- Giao thông giờ cao điểm
- Tuyến đường ô tô trong thành phố
- Xây dựng các tuyến giao thông trong thành phố
- ...

Như vậy thế giới thực hầu như đã, đang và sẽ vẫn xảy ra (thực hiện) song song.

Dĩ nhiên có những vấn đề không thể thực hiện song song. Chẳng hạn không thể xây tường nếu chưa đổ móng, không thể vào cổ áo nếu chưa hoàn thành thân áo, không thể tính số hạng thứ n trong dãy Fibonacci nếu chưa tính các số hạng thứ $n-1$ và $n-2$, chưa thể tính $n!$ nếu chưa tính $(n-1)!$...

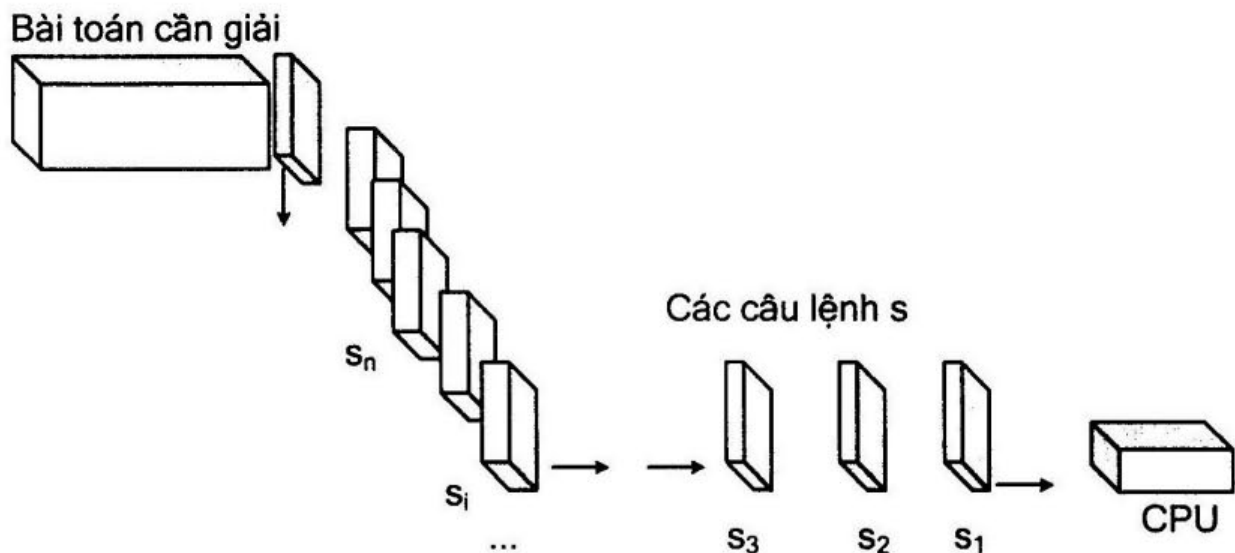
1.1.1. Tính toán tuần tự, chương trình tuần tự và song song

Tính toán tuần tự

- Một vấn đề được chia thành một dãy rời rạc các chỉ thị (Instruction - lệnh máy tính) mà máy tính có thể hiểu để thực hiện.
- Máy tính thực hiện các chỉ thị theo thứ tự mà chúng xuất hiện.
- Chỉ có duy nhất một chỉ thị được thực hiện tại một thời điểm.

Trước khi có các khái niệm như tính toán song song, thuật toán song song, lập trình song song,... ra đời, các phần mềm ứng dụng đều được viết cho các máy tính tuần tự.

Máy tính tuần tự là loại máy tính thực hiện các lệnh một cách tuần tự và tại mỗi thời điểm chỉ một chỉ thị duy nhất được thực hiện – vì vậy máy tính chỉ cần có một CPU (Central Processing Unit – Bộ xử lý trung tâm). Vì chỉ có một CPU nên còn gọi là máy tính đơn nguyên. Cách tính toán tuần tự trên máy đơn nguyên có thể minh họa bởi hình 1.1.



Hình 1.1. Cách giải quyết vấn đề của máy tính tuần tự

Trong hình 1.1, các s_i là các chỉ thị, s_1 xuất hiện trước s_2 nên s_1 được thực hiện trước s_2 , s_2 xuất hiện trước s_3 nên s_2 được thực hiện trước s_3, \dots

Chương trình tuần tự

Chương trình được viết trên một ngôn ngữ nào đó để thể hiện việc tính toán tuần tự được gọi là chương trình tuần tự.

Ví dụ 1.1

Viết chương trình có tên HELLO.CPP (trong C++) để đưa dòng “Hello World !” ra màn hình là chương trình tuần tự. Chương trình có dạng sau đây:

```
s1: #include <iostream. h>
s2: int main ( )
s3: {
```

```
s4: cout << "Hello World ! \n ",  
s5: return 0;  
s6: }
```

Kết quả thực hiện chương trình cho: Hello World !

Rõ ràng là các lệnh của chương trình được thực hiện thứ tự từ lệnh s_1 đến lệnh s_2, \dots , cho đến lệnh cuối cùng là lệnh s_6 .

Tính toán song song

Tính toán song song là quá trình tính toán gồm nhiều tiến trình được kích hoạt đồng thời và cùng tham gia tính toán giải quyết một vấn đề, và nói chung phải thực hiện trên các hệ thống đa bộ xử lý.

Một vấn đề cần tính toán được phân hoạch thành các phần tính toán rời rạc, các phần này lại có thể tính toán được đồng thời (song song).

Từng phần của phân hoạch lại được chia nhỏ thành chuỗi các chỉ thị và được thực hiện tuần tự.

Các chỉ thị của từng phần thực hiện đồng thời trên các CPU khác nhau.

Quan niệm đơn giản nhất, tính toán song song là việc sử dụng đồng thời nhiều máy tính được kết nối thành mạng hoặc chỉ trên một máy tính nhưng có nhiều hơn hai bộ xử lý để tính toán đồng thời.

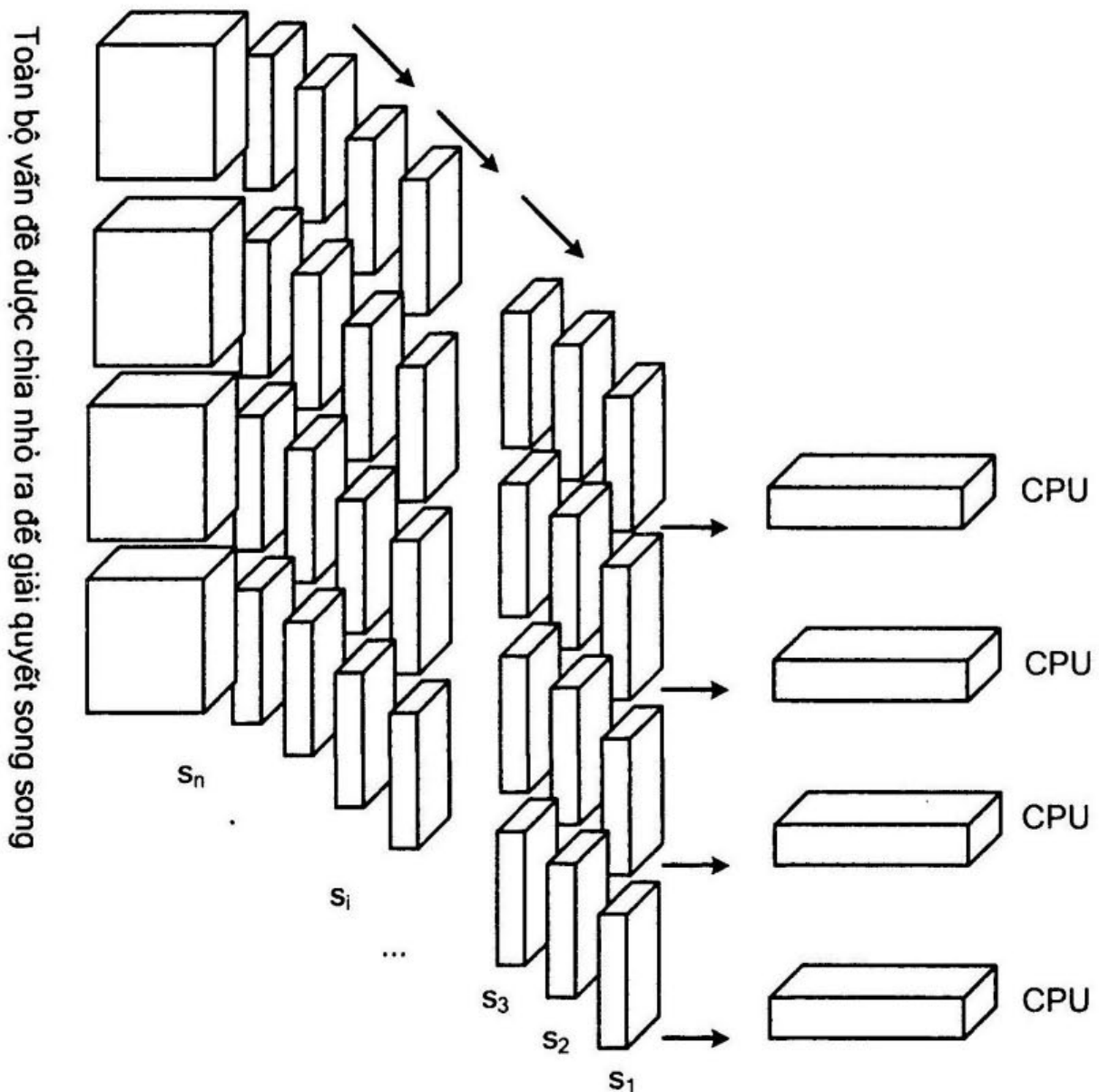
Có thể minh họa tính toán song song bởi hình 1.2.

Như vậy để thực hiện tính toán song song chúng ta cần một trong những tài nguyên tối thiểu sau đây:

- Hoặc một máy tính nhưng có nhiều bộ xử lý,
- Hoặc một số tùy ý các máy tính được kết nối thành mạng;
- Hoặc tổ hợp cả hai loại trên.

Còn các công việc có thể tính toán song song được là những vấn đề có các tính chất sau:

- Có thể chia thành các phần rời rạc, các phần đó có thể thực hiện đồng thời;
- Có thể thực hiện nhiều câu lệnh tại cùng một thời điểm;
- Hiệu năng - theo nghĩa sử dụng ít thời gian (hay tiền bạc) hơn tính toán tuần tự trên máy đơn nguyên.



Hình 1.2. Cách thực hiện các phần song song trên máy tính song song

Chương trình song song

Chương trình được viết trên một ngôn ngữ nào đó để thể hiện việc tính toán song song được gọi là chương trình song song.

Ví dụ 1.2:

Tính tổng hai vectơ: $C = A + B$

Chương trình tuần tự trên máy tính đơn nguyên có thể như sau (giả mã)

```
for(i = 0; i < n; i++)
{
    C[i] = A[i] + B[i];
}
```

Chúng ta sẽ chỉ ra việc song song hóa đoạn chương trình này như sau:

Giả sử có hệ thống đa bộ xử lý gồm p bộ xử lý. Chúng ta có thể phân hoạch n phần tử của hai mảng A và B thành p phần, bộ xử lý thứ k nhận n_k .

Trong đó:

$$k = \overline{1, p}$$

$$n_0 = 0$$

$$\sum_{k=1}^p n_k = n$$

Bộ xử lý	1	2	...	p-1
Các chỉ số phần tử mảng A	0... n ₁ -1	n ₁ ... n ₂ -1	...	n _{p-1} ... n _p -1
Các chỉ số phần tử mảng B	0... n ₁ -1	n ₁ ... n ₂ -1	...	n _{p-1} ... n _p -1
Các chỉ số phần tử mảng C	0... n ₁ -1	n ₁ ... n ₂ -1	...	n _{p-1} ... n _p -1

Cách tính toán viết thành p chu trình sau:

```
for (j = nk-1; j < (nk-1); j++) // (1)
{
    C(j) = A[j] + B[j]
}
```

Trong đó, k là số hiệu của bộ xử lý, chạy từ 1 đến p . Tiến trình (bộ xử lý) thứ k xử lý n_k phần tử.

Hoặc ta có thể cho phép các bộ xử lý truy cập xen kẽ vào các phần tử của mảng như sau:

Bộ xử lý p_k bắt đầu từ phần tử thứ n_{k-1} , sau đó bỏ qua n_k phần tử để xử lý phần tử tiếp theo trong bộ xử lý p_{k+1}, \dots , phân mã chu trình khi đó được viết như sau:

```
for (j=nk-1; j < n; j+ = nk) // (2)
{
    C[j] = A[j] + B[j];
}
```

Các chu trình ở (1) hoặc (2) đều có thể thực hiện song song.

Trong trường hợp p bộ xử lý giống nhau, n chia hết cho p thì phương pháp chia các phần tử là chia đều cho các bộ xử lý; nghĩa là mỗi bộ xử lý có $\frac{n}{p}$ phần tử.

Mục đích của tính toán song song

Về phương diện lịch sử, tính toán song song được nghiên cứu để nâng cao hiệu năng tính toán, và được sử dụng để mô hình hóa các vấn đề khoa học kỹ thuật rất phức tạp trong thế giới thực. Để minh chứng cho vấn đề trên chúng ta có thể xét một số ví dụ mô hình hóa các vấn đề song song sau đây: