**Rate Monotonic Scheduling – Lập lịch tỷ lệ đơn điệu**

1. **Khái niệm cơ bản**

Rate Monotonic (RM) là một thuật toán trưng dụng, lập lịch tĩnh trên hệ thống bộ xử lý đơn.

Các tác vụ định kỳ (periodic task) T được định nghĩa như sau:

T = (C, P, D)

Với C là thời gian cần thực thi, P là chu kỳ và D là hạn định của T.

Phương pháp lập lịch này dựa trên một số giả thiết sau đây:

* Tất cả các tác vụ tham gia hệ thống phải có hạn định (deadline) kiểu chu kỳ.
* Thời gian thực hiện của các tác vụ biết trước và không đổi.
* Hạn chót cho các tác vụ là vào cuối mỗi chu kỳ (D = T).
* Các tác vụ phải sẵn sàng thực hiện vào đầu mỗi chu kỳ.
* Các tác vụ độc lập với nhau, nghĩa là không có quyền ưu tiên giữa các tác vụ và chúng không chặn lẫn nhau.
* Chi phí cho chuyển đổi ngữ cảnh, hoán đổi, v.v... rất nhỏ và có thể bỏ qua.

Thuật toán RM dựa trên độ ưu tiên. Dựa trên chu kỳ, tác vụ nào có chu kỳ càng nhỏ thì mức ưu tiên càng cao. Điều này có nghĩa rằng, hạn định của mỗi tác vụ là không đổi.

Để tập các tác vụ đã cho có thể lên lịch, phương trình sau cần được thỏa mãn:

Trong đó:

* U là mức sử dụng của bộ xử lý.
* n là số lượng tác vụ.
* thứ tự là thời gian cần thực thi và thời gian chu kỳ của tác vụ thứ i.

1. **Cài đặt**

Cấu trúc chương trình được cài đặt như sau:

1. find\_lcm(x, y)

* Tìm bội chung nhỏ nhất của chu kỳ thời gian x và y (giây).

1. rm\_schedule(tasks, duration=None)

* Hàm chính, lập lịch cho các tasks.
* Sắp xếp các tasks theo thứ tự tăng dần về chu kỳ.
* Chọn ra và cập nhật các tasks được đem vào quá trình xử lý.

1. draw\_timing\_diagram(timeline, task\_list)

* Vẽ biểu đồ minh họa.

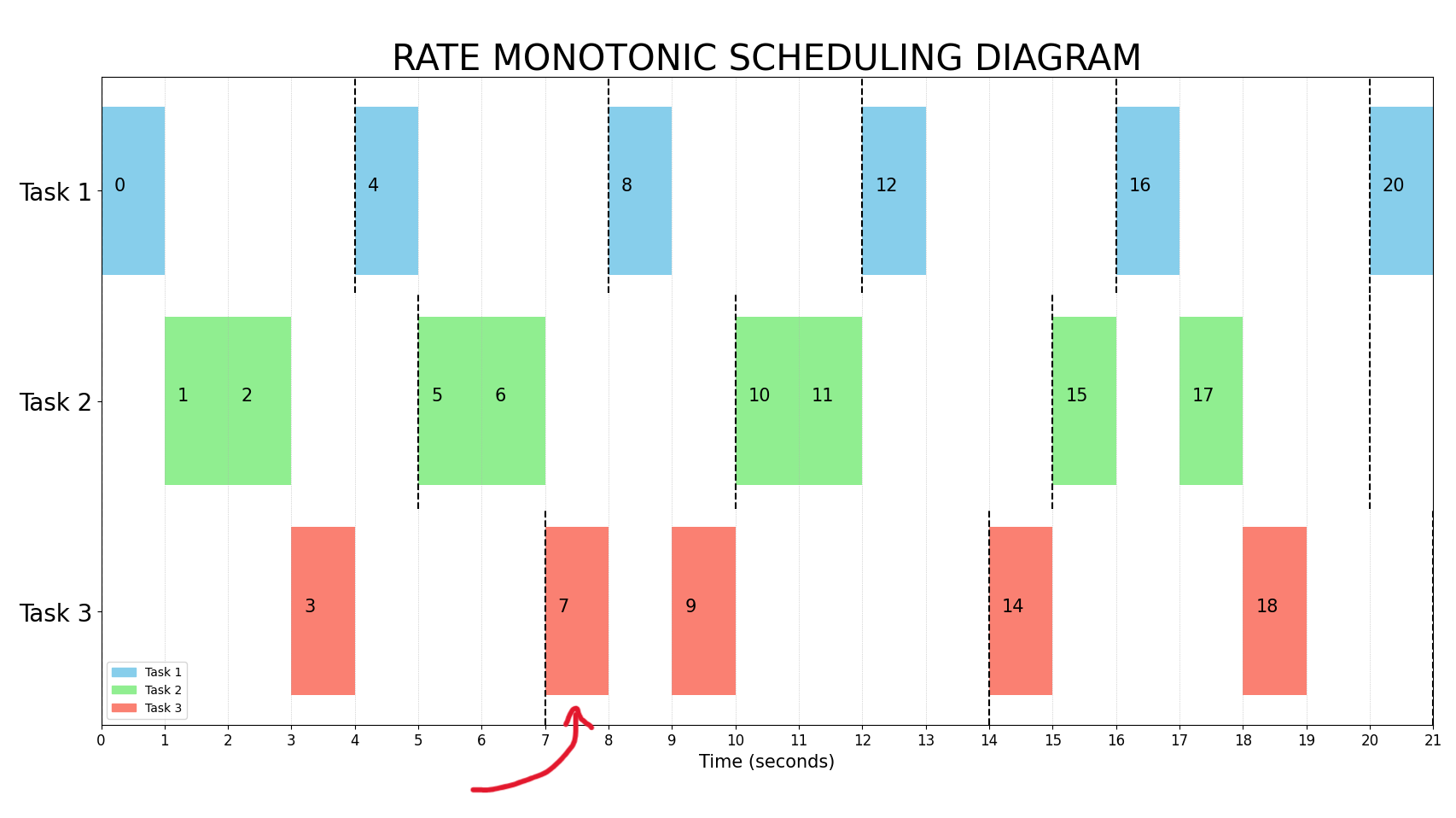
1. **Ví dụ minh họa**

***Ví dụ 1:*** Xét 03 tác vụ được xác định theo bảng sau:

|  |  |  |  |
| --- | --- | --- | --- |
| Tác vụ | Thời gian thực hiện | Chu kỳ | Hạn định |
| T1 | 1 | 4 | 4 |
| T2 | 2 | 5 | 5 |
| T3 | 2 | 7 | 7 |

Ta có: U = 1/4 + 2/5 + 2/7 = 0.9357 > 0.7977 =

Sơ đồ thời gian làm việc của bộ xử lý:



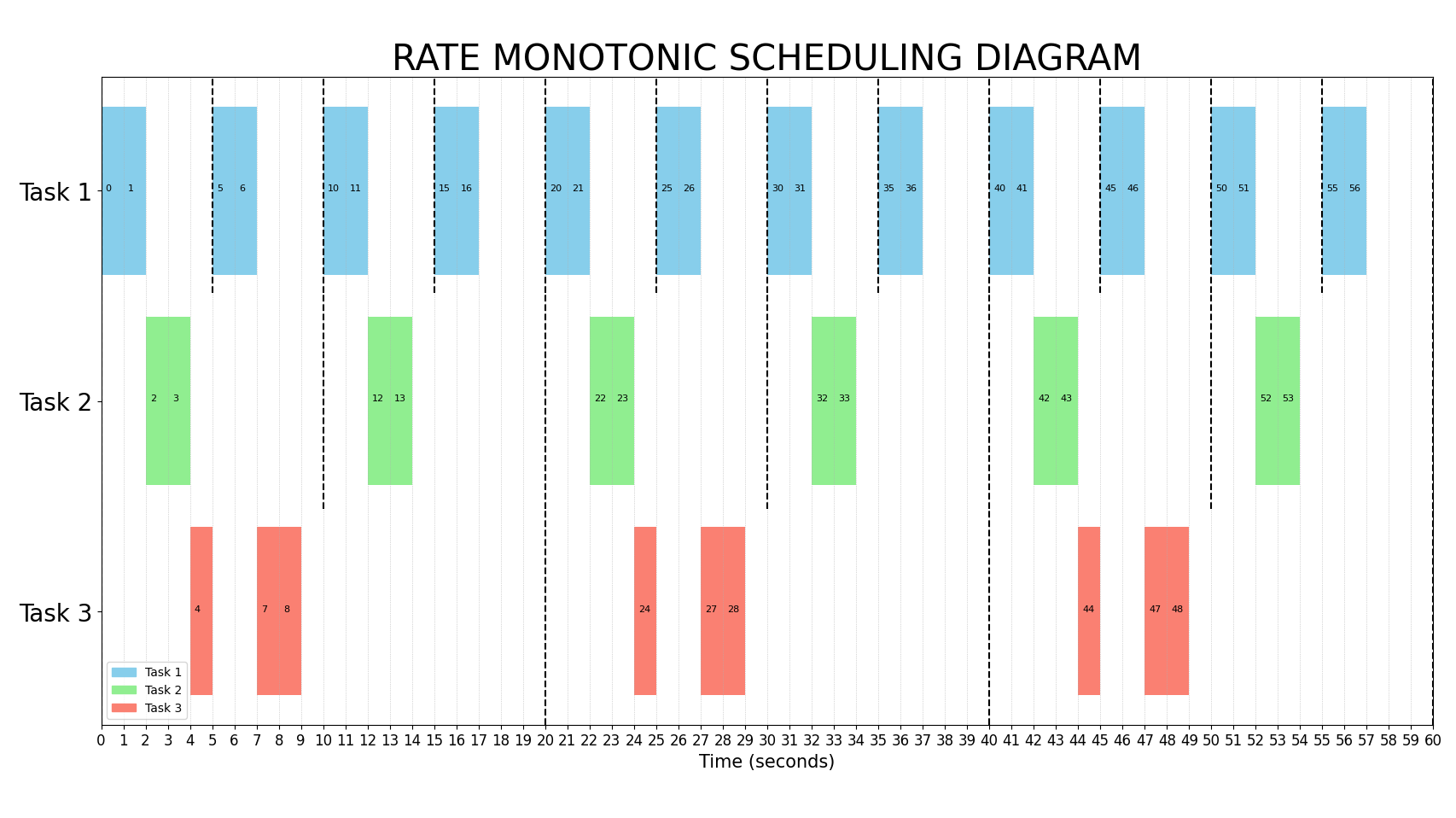
Nhận xét: Hạn định trễ ở vị trí số 7 (mũi tên trong hình) do điều kiện [1] không được thỏa mãn.

***Ví dụ 2:*** Xét 03 tác vụ được xác định theo bảng sau:

|  |  |  |  |
| --- | --- | --- | --- |
| Tác vụ | Thời gian thực hiện | Chu kỳ | Hạn định |
| T1 | 3 | 20 | 20 |
| T2 | 2 | 5 | 5 |
| T3 | 2 | 10 | 10 |

Ta có: U = 3/20 + 2/5 + 2/10 = 0.75 < 0.7977 =

Dựa vào độ ưu tiên về thời gian của chu kỳ, T1 → Task 3, T2 → Task 1, T3 → Task 2.



Nhận xét: Không có hạn định trễ, do điều kiện [1] đã được thỏa mãn.

1. **Nhận xét thuật toán**
2. Ưu điểm

* Cài đặt tương đối dễ dàng.
* Nếu tồn tại bất kỳ thuật toán gán ưu tiên tĩnh nào có thể đáp ứng thời hạn thì RM cũng có thể làm điều tương tự. Thuật toán này là tối ưu.

1. Nhược điểm

* Rất khó để hỗ trợ các tasks rời rạc và không định kỳ bằng RM.
* RM không tối ưu khi thời gian thực hiện và thời hạn khác nhau.

1. **Tham khảo**

[1] – International Journal of Computer Applications – Volume 158 – No 6, January 2017, *“Comparison of Real Time Task Scheduling Algorithms”.*

[2] – GeeksForGeeks – *Rate-monotonic scheduling*

<https://www.geeksforgeeks.org/rate-monotonic-scheduling/>

Truy cập lần cuối: 19/05/2025.