**系统任务调度策略优化**

# 背景

在电信运营级别路由器、交换机设备中，主控单板（使用多核处理器，运行Linux操作系统）承担了重要的设备管理、协议支持等工作（如大量的进程/线程，用于RIP/OSPF/BGP/ISIS等路由协议的处理）；一般情况下都有上百个进程、几千个线程在运行，但并非所有线程都忙；视开通的业务不同，某一时间阶段内只有部分业务线程忙。设备管理类进程（比如设备电源监控、环境温度监控、风扇调速）优先级不高，但必须在一个相对较长的周期内（一般在分钟级别）得到调度。

对系统存在如下要求：（1）业务类线程优先级高，一旦请求到来必须立即响应，如网络拓扑发生变化后路由表更新等；（2）设备管理/状态监控线程不需要立即响应，但是必须在一段时间内运行；（3）一些辅助线程，如日志回写等，工作在空闲时刻即可。（4）在系统当中，工作线程并非都是独立的，A线程输出为B线程输入，B线程的输出会作为C线程的输入，A/B/C线程优先级并不相同；诸如此类的多线程协同设计非常常见。

系统对稳定性要求很高，高并发处理时系统不能发生死锁等情况，低优先级进程不能“饿死”；整体性能要求高，必须做到有事务才占用处理器资源，无事务不占用资源。

# 现有解决方案

利用当前Linux已有的调度策略，业务线程采用实时优先级策略，其他线程采用普通优先级策略，我们可以基本达成设计目标。

# 问题&改进方案

然而仍然存在一些缺陷。如，持续的公共网络局部拓扑变化会导致路由持续有大量的更新，致使业务进程/线程繁忙，这些进程导致CPU高占用影响设备运行、甚至发生失去响应、死锁等情况。

1. 简述当前Linux操作系统支持哪几类调度算法？这些调度算法各有什么优缺点？除了这些算法，你是否还了解有影响力但未被Linux主线纳入的调度算法？尽可能多的呈现有关调度算法的实现细节/特点。
2. 基于当前的Linux操作系统（4.x及之上版本）提供的调度算法，请你设计整个系统中各应用线程的调度策略，能够更好的提升系统性能、稳定性。
3. 如果需要你根据当前系统应用情况，定制专用调度算法，你会做哪些方面的改进，以更好的提升性能和稳定性。

要求：请您以Word输出整体运作方案，并将其中要点以PPT形式进行输出，在极致挑战环节进行宣讲。