

1)

Se podría hacer con una única cola. En cada iteración, si justo se desocupó uno o varios núcleos, se debería extraer elementos READY de la cola hasta que todos los núcleos tengan un proceso RUNNING. Se escoge cualquiera que esté desocupado. Si hay un interrupt se coloca el proceso actual al final de la cola y el primero de la cola pasa a READY en el núcleo en que ocurrió la interrupción. En general se trata de tener todos los núcleos ocupados cuando hay procesos en cola. Habría que evaluar si algunos procesos son dependientes de otros, y si por lo tanto debe haber un orden secuencial.

No mejoraría el turnaround time pues cada proceso aún estaría siendo procesado por un sólo núcleo, y por lo tanto el proceso en sí tomaría la misma cantidad de tiempo que en un sistema de un sólo núcleo.

El waiting time disminuiría porque la cola avanzaría más rápido, al distribuirse los procesos entre los distintos núcleos.

El response time sería menor dado que al disminuir el waiting time (con menos tiempo en cola) el tiempo entre el fin de la entrega de input y el output, que fundamentalmente corresponde al tiempo en cola, debería decrecer.

2)

El $O(1)$ es un sistema en que cada proceso obtiene una cantidad constante de tiempo, independiente de cuantos procesos se esté corriendo. Se ocupó en Linux, en el kernel 2.6. Una de las características que lo hace atractivo es que tiene una cota superior para el tiempo de ejecución, lo cual es particularmente útil en el ámbito del "real-time", en que se busca ejecución determinista. Dado que garantiza ejecutar cada proceso antes de una cantidad constante de tiempo, puede manejar una cantidad muy grande de tareas (uso intensivo del cpu) sin incrementar "overhead costs" a medida que aumenta la cantidad de procesos en cola. Es un scheduler interactivo dado que determina a partir de una heurística qué procesos son interactivos (esperan input de un usuario durante horas, por ejemplo) y les asigna una prioridad más alta.