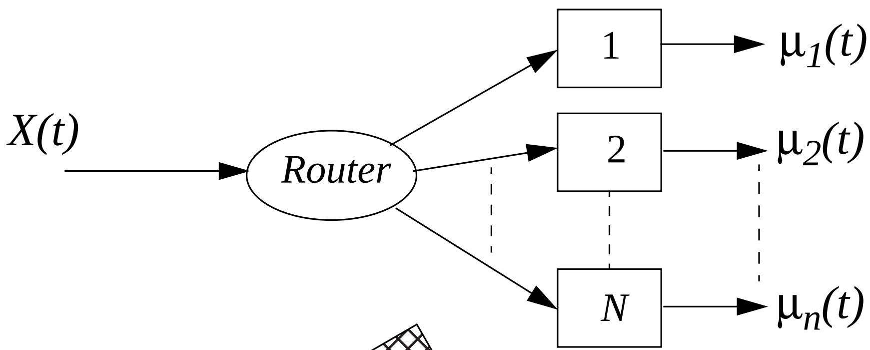
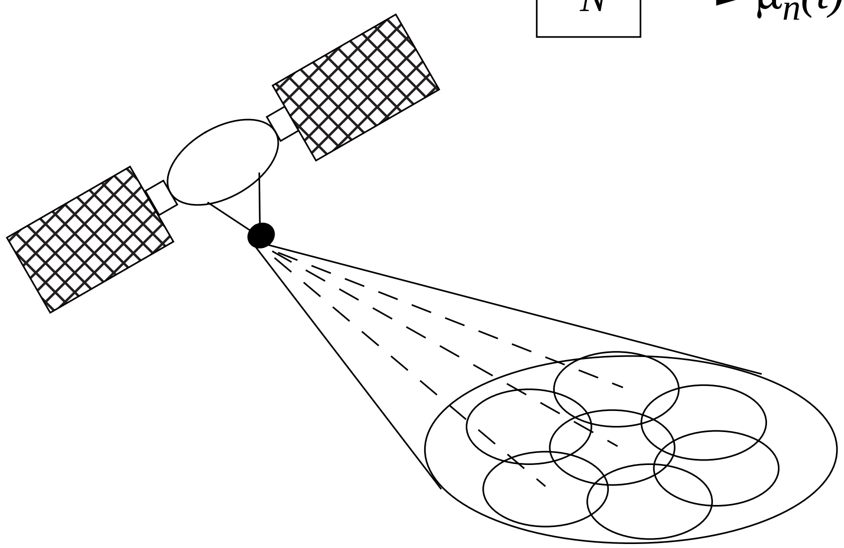
Packet Routing over Parallel Time-Varying Queues with Application to Satellite and Wireless Networks



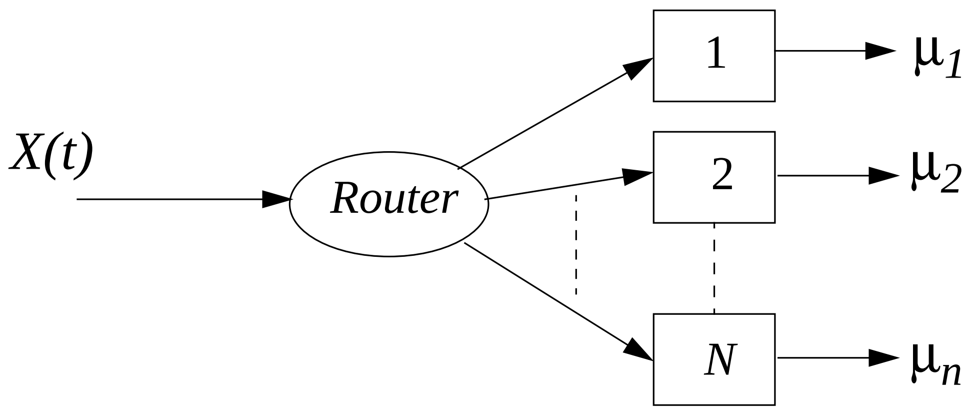


Michael Neely
MIT - LIDS
mjneely@mit.edu

Eytan Modiano (LIDS)
Charlie Rohrs (LIDS)

M.J. Neely, E.Modiano, and C.E.Rohrs, "Dynamic Routing to Parallel Time-Varying Queues with Applications to Satellite and Wireless Networks," Conference on Information Sciences and Systems, Princeton University: March 2002.

Consider a constant service rate routing problem:
(heterogeneous service rates $\{\mu_1, \mu_2, ..., \mu_n\}$)



2 Natural Routing Strategies:

Greedy: π_{greedy}

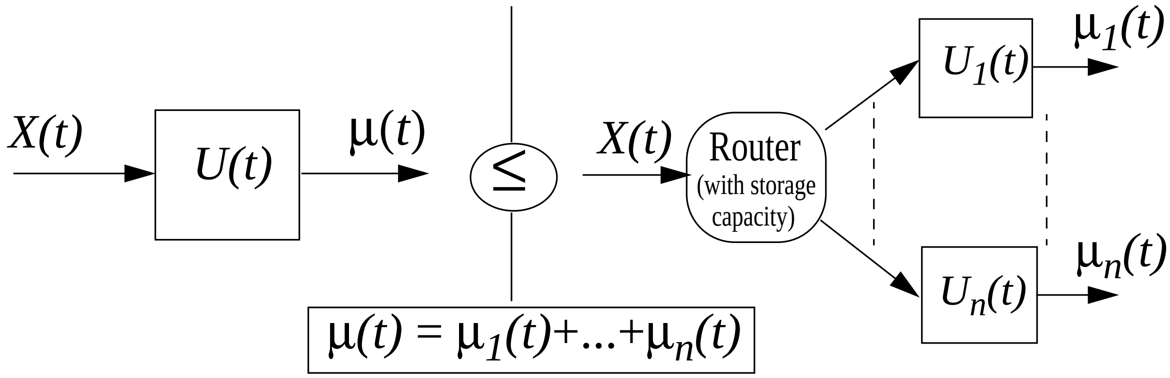
Choose queue k such that $k = \underset{j \in \{1, \ldots, n\}}{argmin} \left\{ \frac{L_i + U_j(t)}{\mu_j} \right\}$.

Work Conserving: π_{WC}

Choose queue k such that $k = \underset{j \in \{1, \ldots, n\}}{argmin} \left\{ \frac{U_j(t)}{\mu_j} \right\}$.

$U_{greedy}(t)$ can be arbitrarily larger than $U_{WC}(t)$. However, $U_{WC}(t)$ stays within a fixed upper bound from any other strategy.

Multiplexing Inequality:



$$U_{single}(t) \le U_{multi}(t)$$ (For any routing strategy over the parallel queues)

However, for the work conserving strategy π_{WC}, we also have an upper bound:

$$U_{single}(t) \le U_{WC}(t) \le U_{single}(t) + (n-1)L_{max}$$

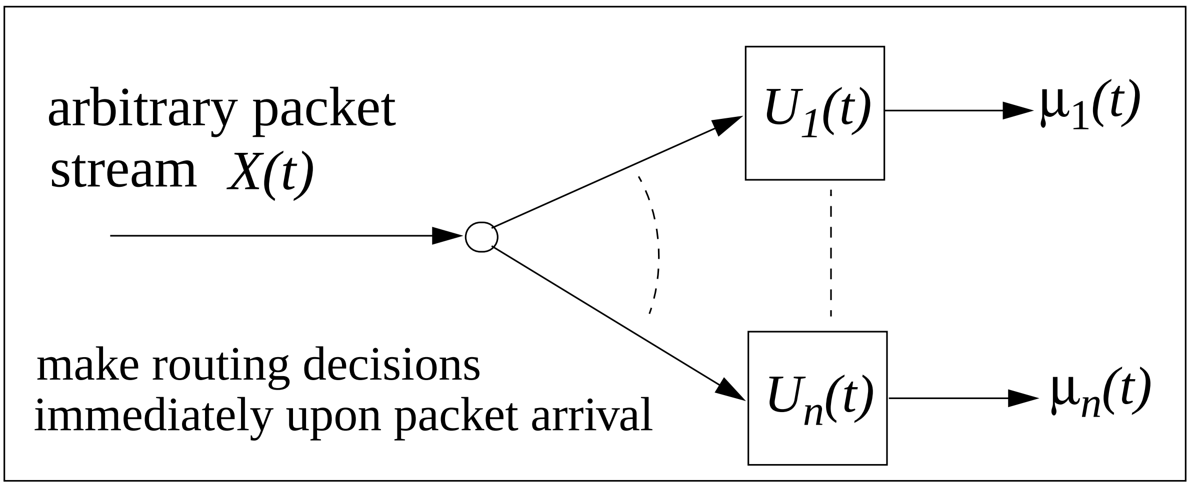
Comparing π_{WC} to any other routing strategy π:

$$U_{WC}(t) \le U_{\pi}(t) + (n-1)L_{max}$$

...and it can be shown that $(n-1)L_{max}$ is the best bound possible for non-predictive, non-preemptive routing schemes, hence π_{WC} is *minimax optimal*.

The π_{WC} routing algorithm uses a pre-queue to achieve work conservation in systems with time-varying server speeds (route to a server immediately when it empties).

How do we route when no pre-queue is available?
(Ex: Queues are in different physical locations)



Input process $X(t)$ --- rate ergodic, rate λ.

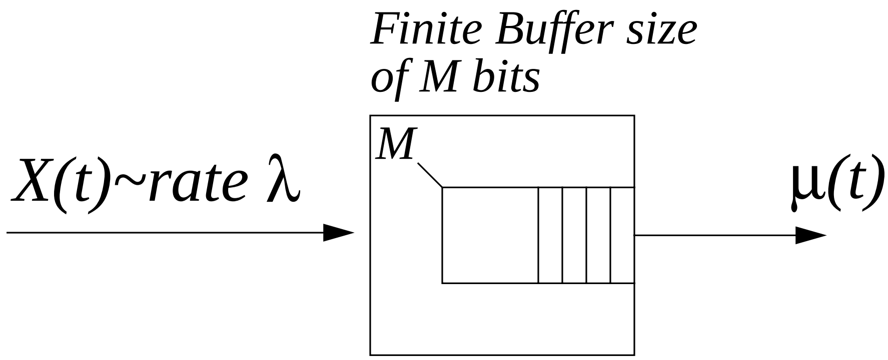
Processing rates $\{\mu_i(t)\}$ --- ergodic, time average rates $\{\mu_i^{av}\}$.

How do we stabilize the system without knowing the input stream, and without knowing future processing rates?

Consider Join-the-Shortest-Queue strategy: π_{JSQ}
(JSQ = Route the incoming packet to the queue j with the smallest unfinished work $U_j(t)$).

New notion of stability useful for understanding stability issues in systems with general ergodic inputs:

Consider a single server queue with a finite buffer of size *M*:



Define *DR(M)* = Packet drop rate when buffer size is *M* bits. (clearly *DR(M)* is a non-increasing function of *M*).

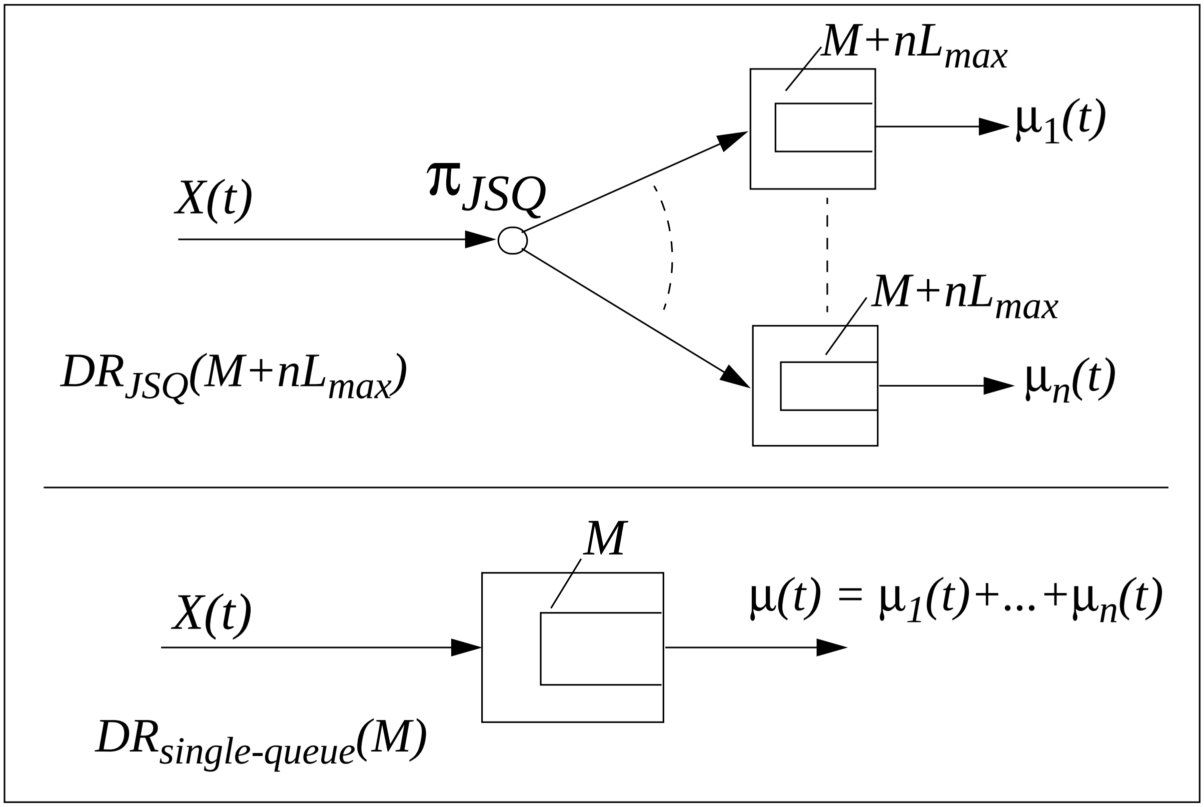
Definition:
A system is *loss rate stable* if $DR(M) \to 0$ as $M \to \infty$.

This definition is closely related to the existing notion of stability defined in terms of a vanishing complementary occupancy distribution $Pr[U > m] \to 0$ as $m \to \infty$. It can be shown:

$\lambda \le \mu_{av}$: necessary condition for stability.

$\lambda < \mu_{av}$: sufficient condition if inputs and linespeeds are Markov Modulated.

Compare drop rate under *JSQ* policy to a single-server queue:



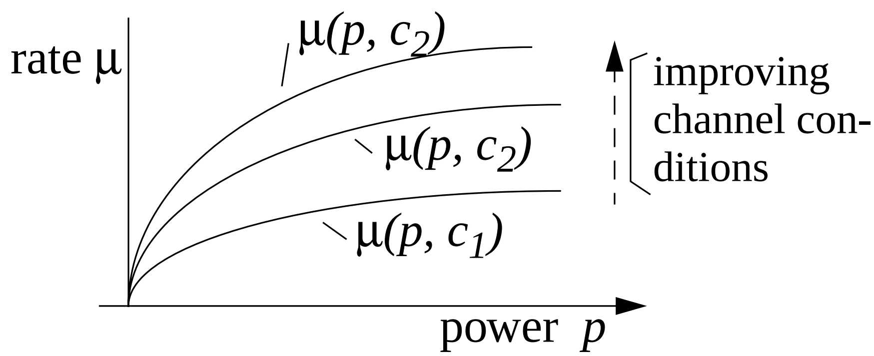
Let $DR_{JSQ}(M)$ represent the packet drop rate in the multi-queue system under the *JSQ* routing policy when all queues have buffer size *M*.

Theorem:

$$DR_{JSQ}(M+nL_{max}) \leq DR_{single\text{-}queue}(M)$$

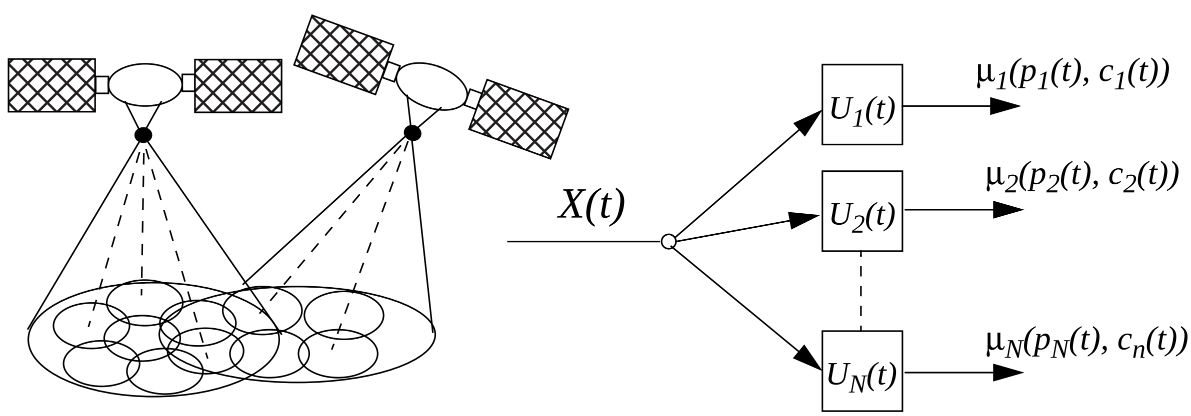
Thus, the system under π_{JSQ} is loss rate stable <u>iff</u> the single queue system is loss rate stable.
(Hence, it is stable whenever the system is stabilizable).
Joint routing and Power Allocation:

Power Allocation--Processing rates depend on power allocation $p_i(t)$ and time varying channel state $c_i(t)$: $\mu_i(p_i(t), c_i(t))$.



Each satellite s has multiple beams and a fixed power resource $P_{tot}^{(s)}$.

Must jointly route packets and allocate power to the different queues subject to a fixed power resource $\sum p_i(t) \le P_{tot}$.



Decoupled Policy:

-Routing: JSQ

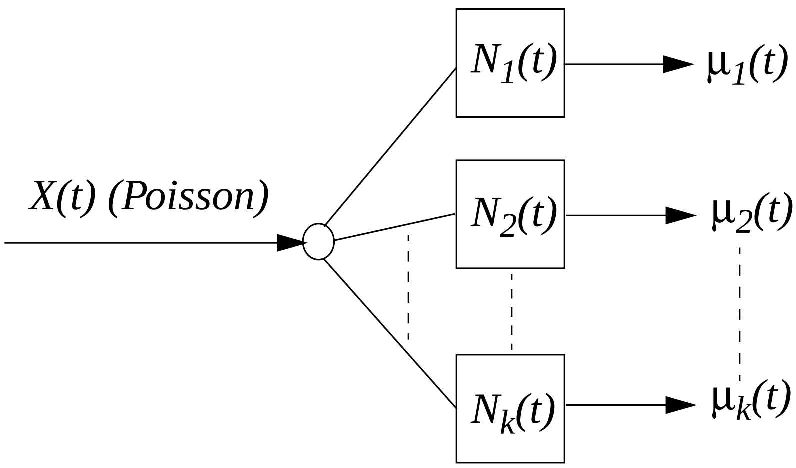
-Power Allocation:

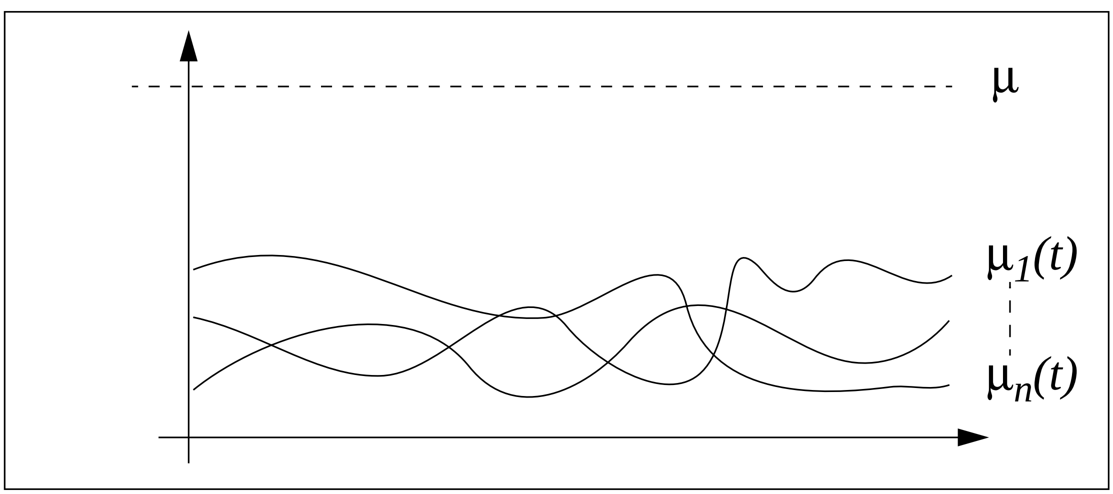
Maximize $\sum \mu_i(p_i, c_i(t))$ subject to $\sum p_i = P_{tot}$

Example: Poisson arrival process, fixed length packets (size L).

Assume, for the simplicity of the example, that the time varying linespeeds $\mu_i(t)$ are arbitrary but sum to a constant rate μ.

Let $N_i(t)$ = Number of packets in queue i at time t.





Translate unfinished work into number of packets: $N = \lceil U / L \rceil$

$$DR_{JSQ}(M) \leq DR_{Single}(M-k) \leq Pr[N_{M/D/1} > n]$$

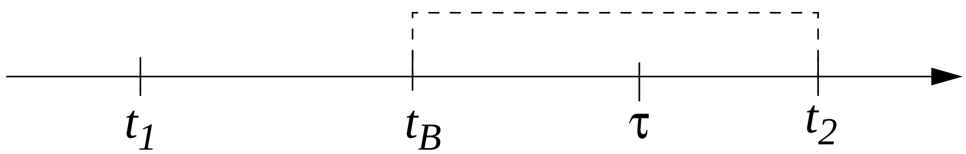
Theorem:

$$DR_{JSQ}(M+nL_{max}) \leq DR_{single\text{-}queue}(M)$$

Proof outline: Let $G(t)$ represent packet drops during [0, t].

We show $G_{JSQ}(t) \leq G_{single}(t)$ for all time t.

Prove claim over "*completely busy periods*":



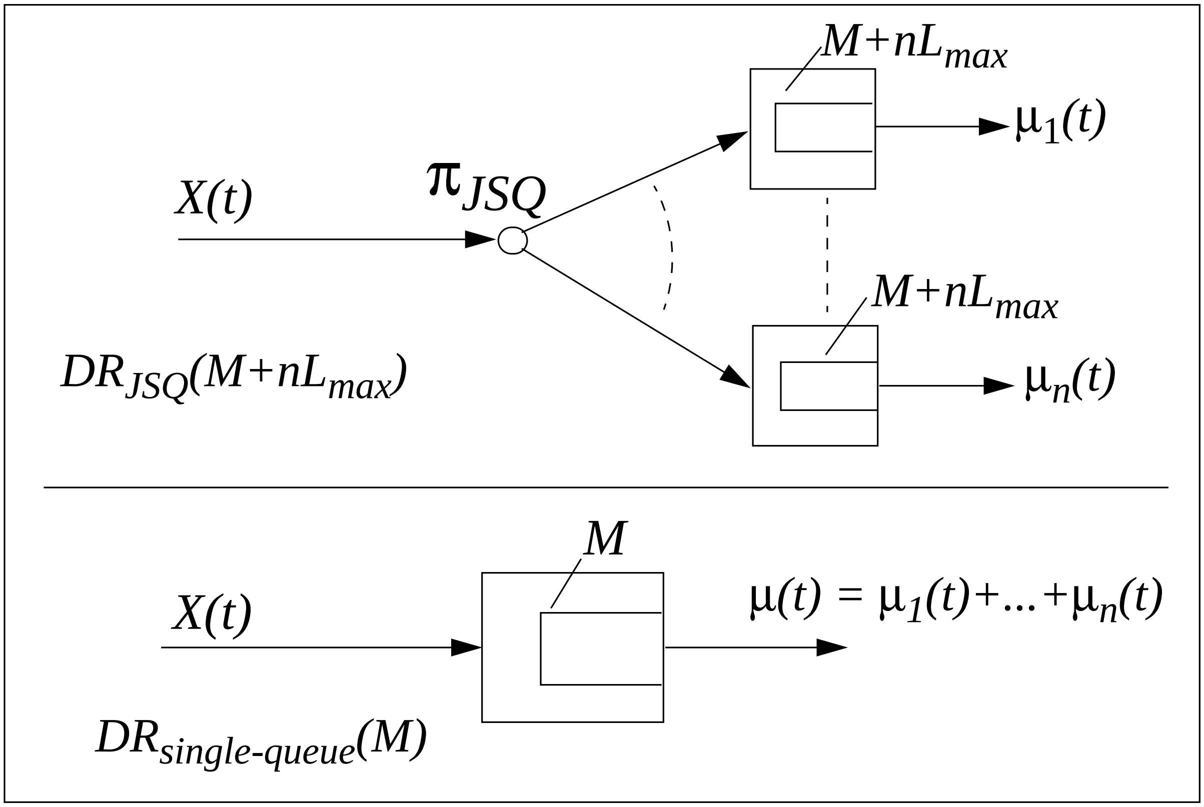
Let: a = arrivals during [t_B, t].
d = departures during [t_B, t].

1. Packet Conservation equalities:

$$U_{JSQ}(\tau) = U_{JSQ}(t_B) + a - d_{JSQ} - g_{JSQ}$$

$$U_{single}(\tau) = U_{single}(t_B) + a - d_{single} - g_{single}$$

2. $d_{JSQ} \geq d_{single}$.



3. Just before c.b.p., at least one queue of multi-server system is empty:

$$U_{JSQ}\left(t_B\right) \leq (n-1)[M + nL_{max}]$$

4. JSQ Strategy: When a packet is dropped at time τ, all queues must have more than $[M+(n-1)L_{max}]$ unfinished work:

$$U_{JSQ}(\tau) > n[M + (n-1)L_{max}]$$

These facts plus algebra yield the result. ❑