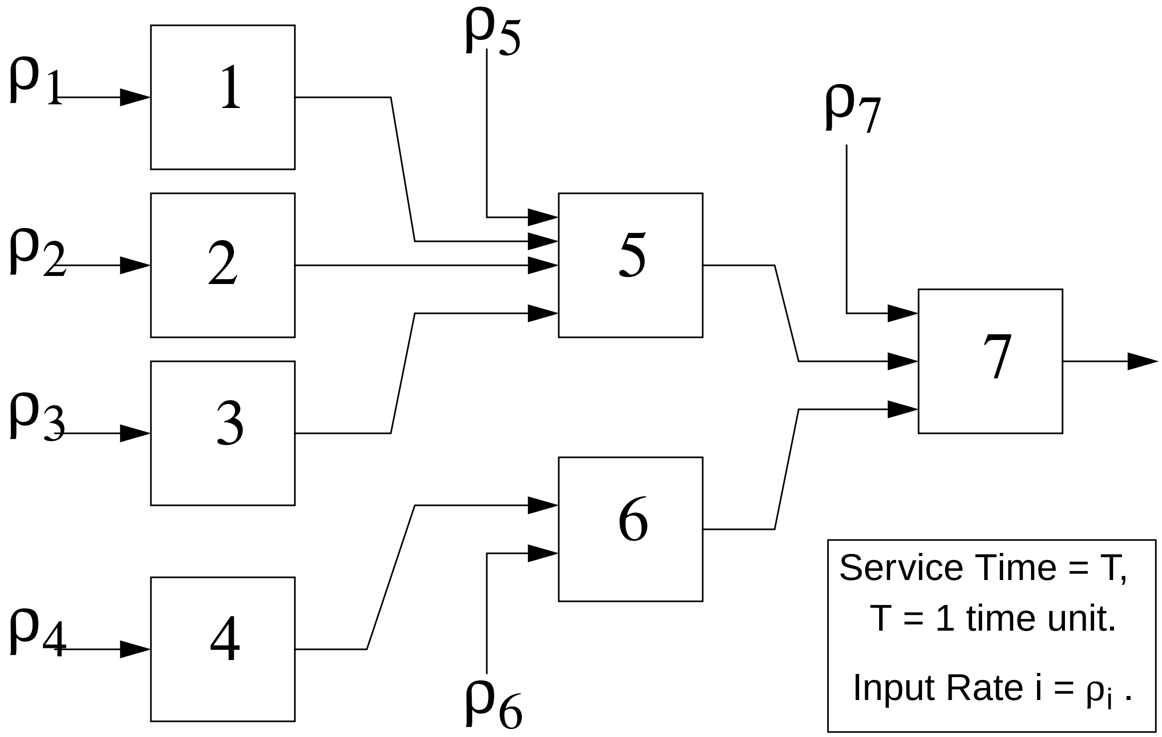
Analysis of Multi-Stage Tree Networks of Deterministic Service Time Queues

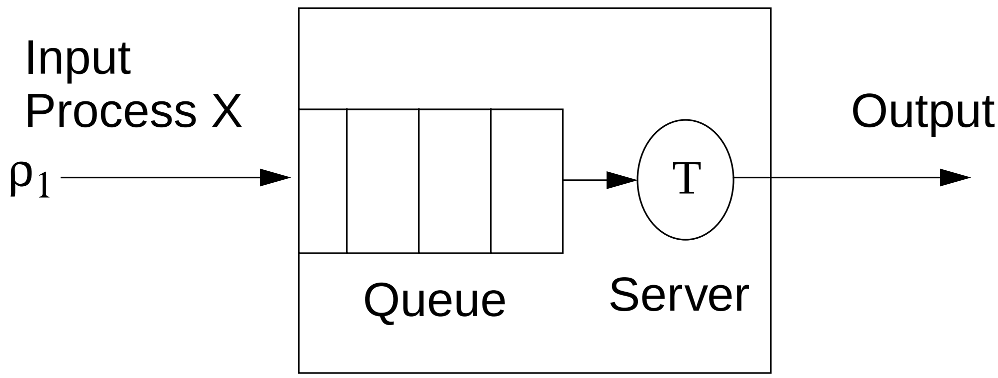


Question: Can we analyze such systems without making any assumptions about the underlying exogenous input processes?

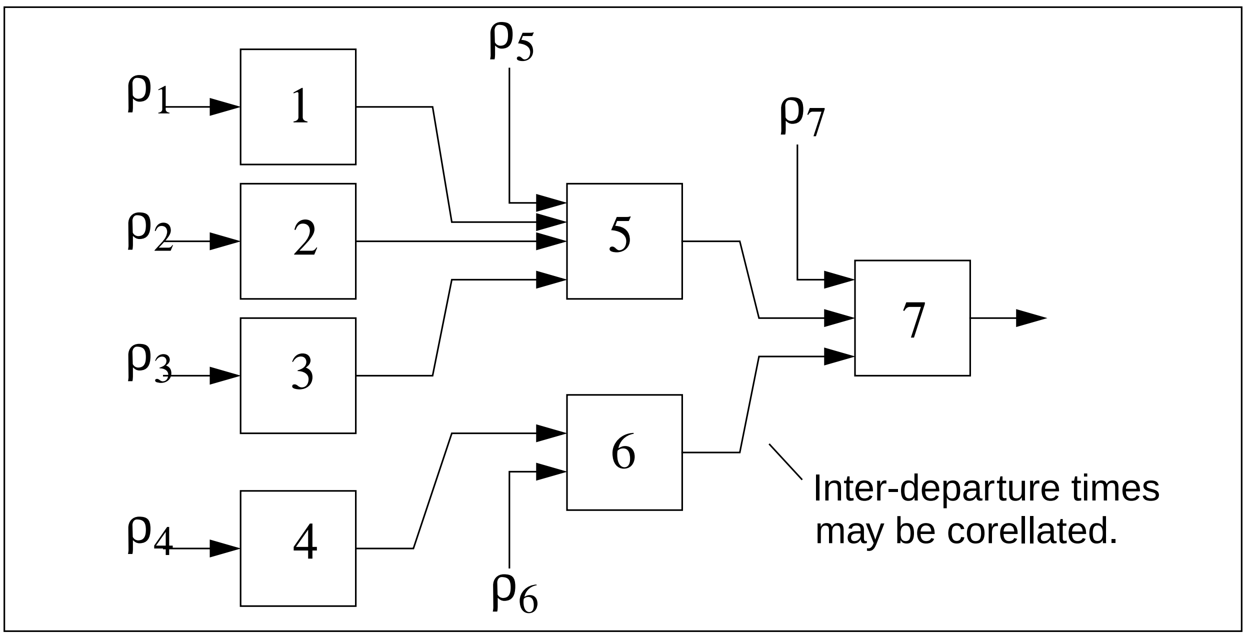
Mike Neely
MIT -- LIDS mjneely@mit.edu
Advisor: Charlie Rohrs

M.J. Neely, C.E. Rohrs, "Equivalent Models and Analysis for Multi-Stage Tree Networks of Deterministic Service Time Queues" Proceedings of the 38th Annual Allerton Conference on Communication, Control, and Computing, Oct. 2000.

Single Stage System:



Multi-Stage Tree System:

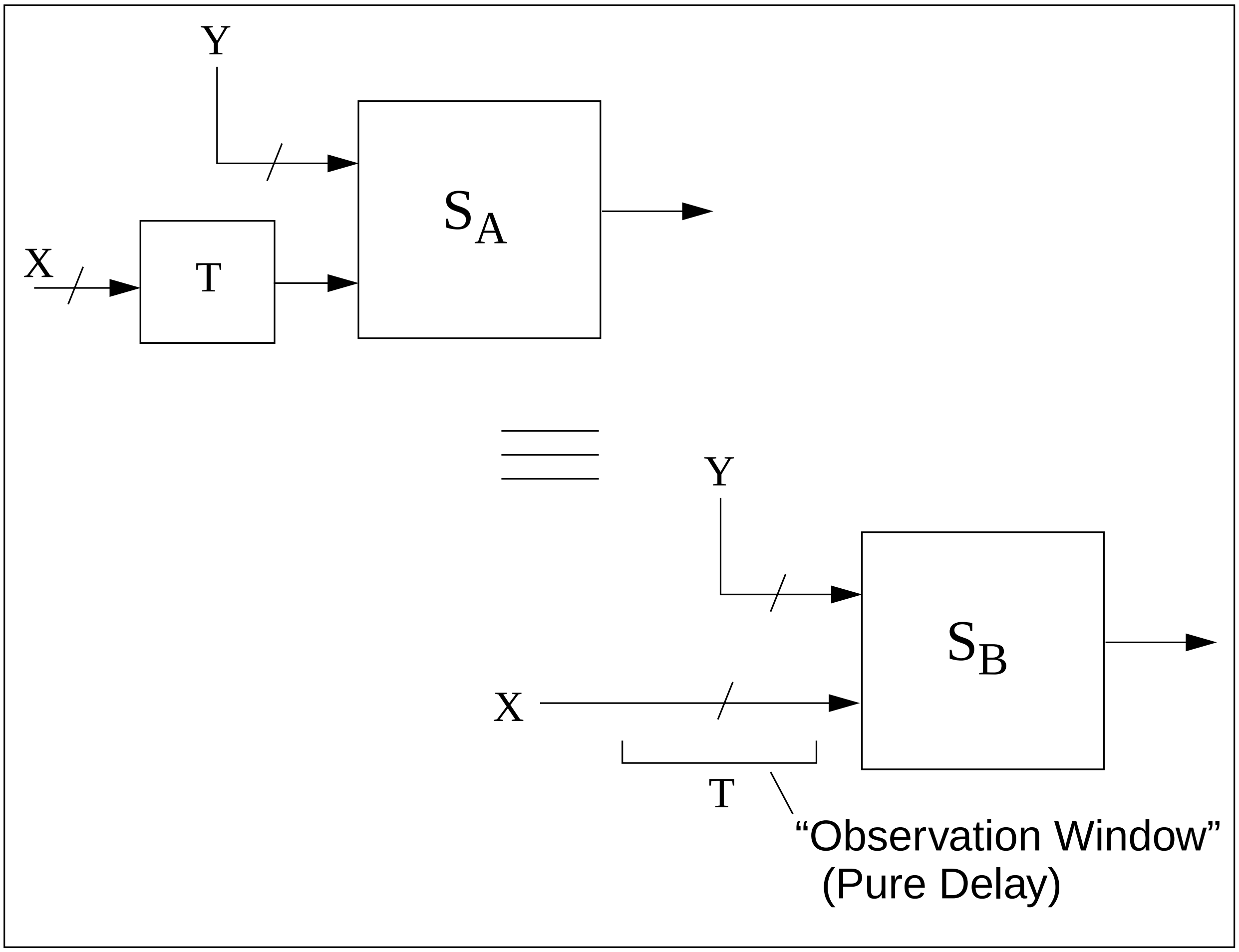


ATM chops packets into fixed length "Cells," creating a Non-Jacksonian System with Deterministic Service Time T.

Thus: Inter-departure times may be corellated.

"**Theorem 2**": *Equivalence theorem*:*
The 2 systems below are "equivalent."

*A similar statement was published independently and previously by Shalmon and Kaplan [1].



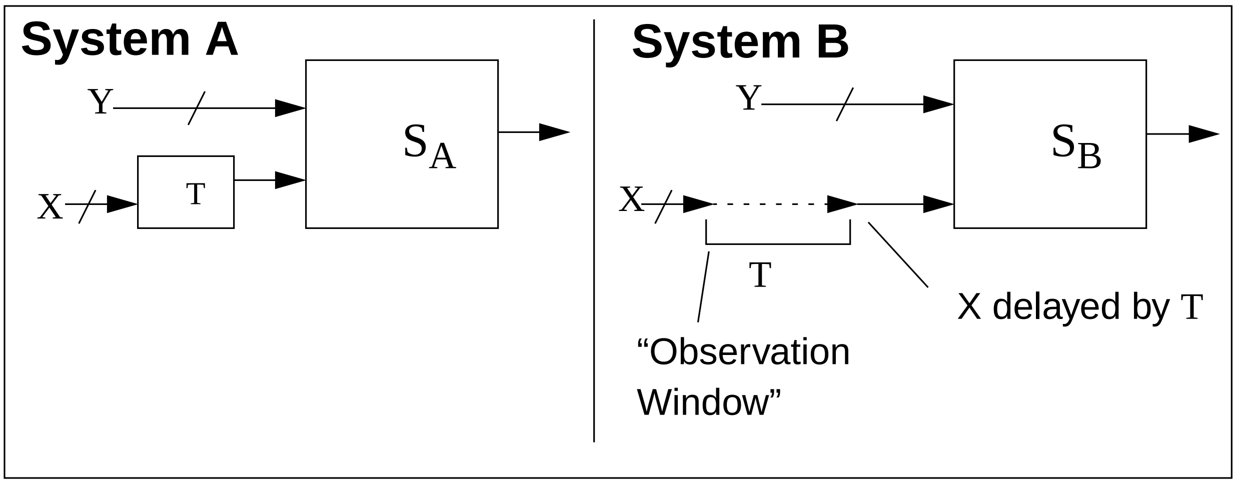
If all packets have deterministic service times T:

1) S_A empty *iff* S_B empty. (Same busy/idle periods).

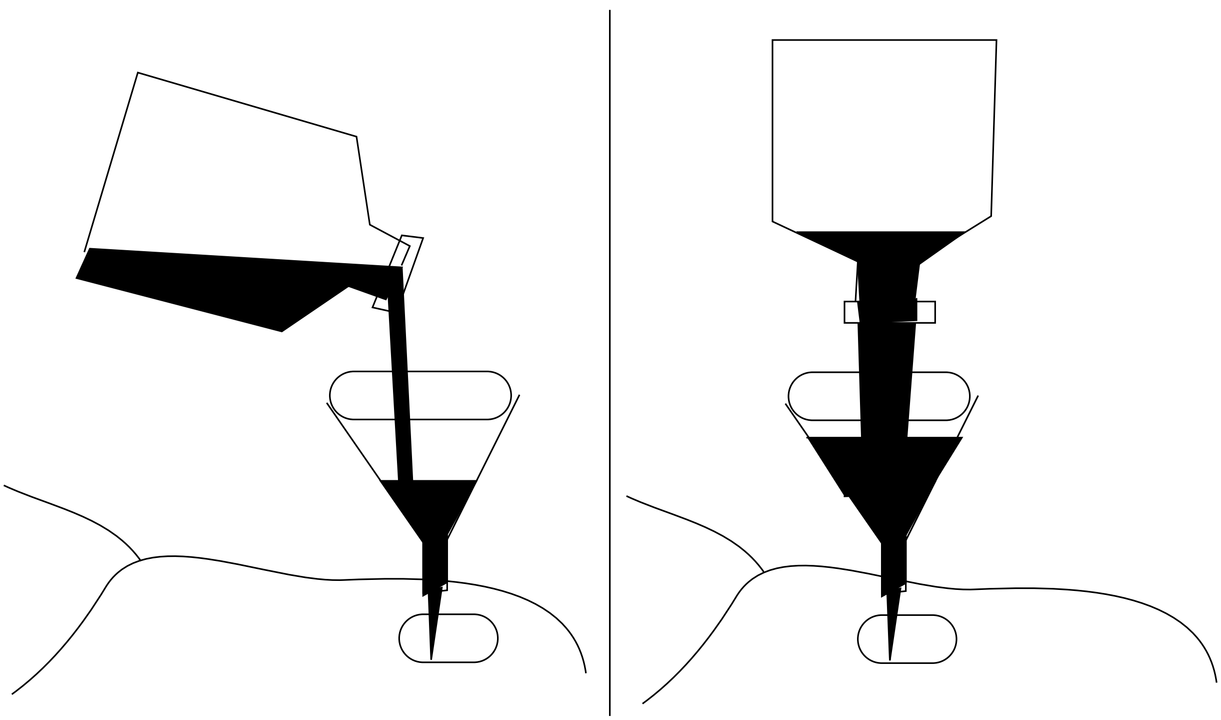
2) Entire system A empty *iff* entire system B empty.

3) Departures from systems A and B are the same, and hence the number of packets within the entire systems A and B are equal at every instant of time.

Caveat: The *ordering* of packets may be shuffled, but the number of packets in both systems will always be the same.

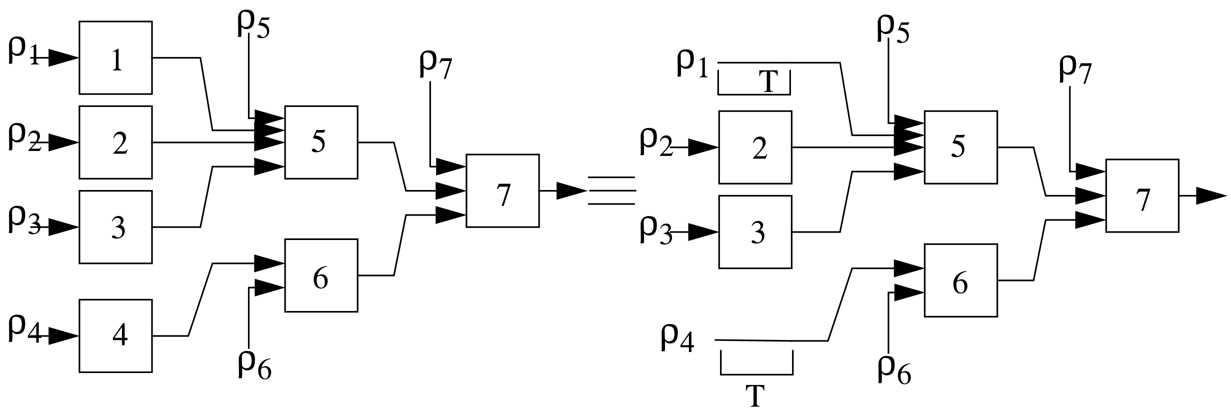


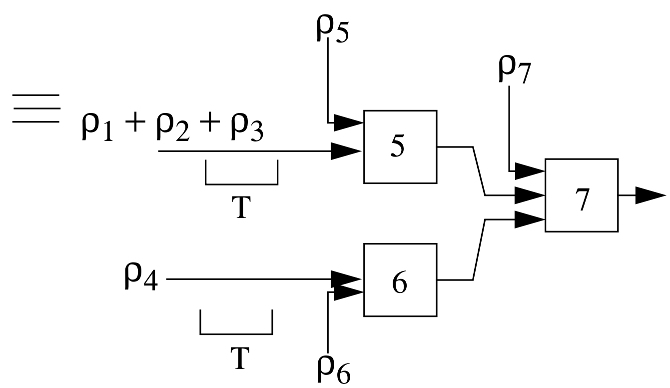
Intuition: Changing your oil...



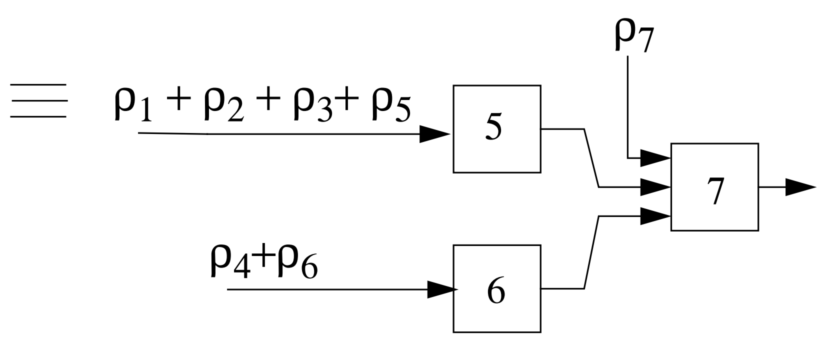
The manner in which we pour oil from the bottle to the funnel does not affect the departure process from the funnel as long as the funnel is not empty.

Use Theorem 2 iteratively on multi-stage systems. Analyzing the last node, we find:





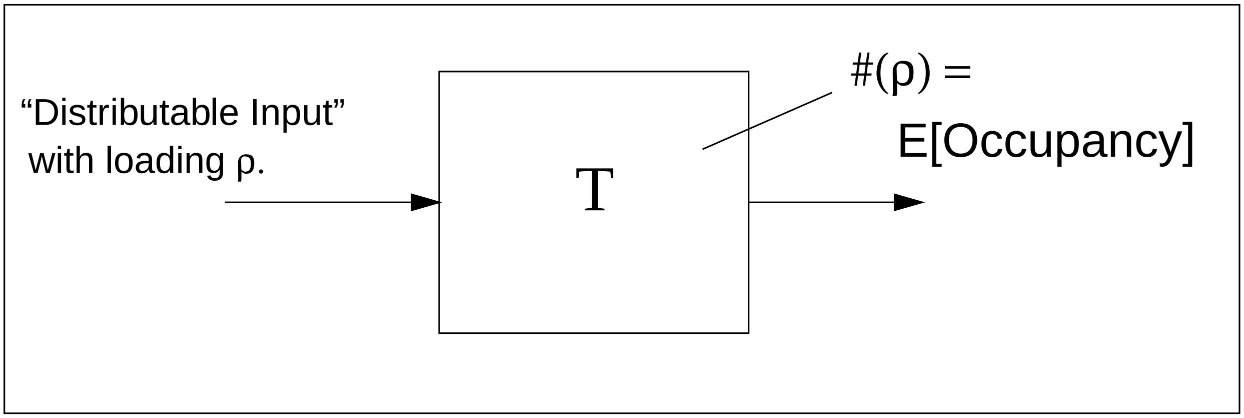
If inputs are independent and stationary, time delays don’t matter:

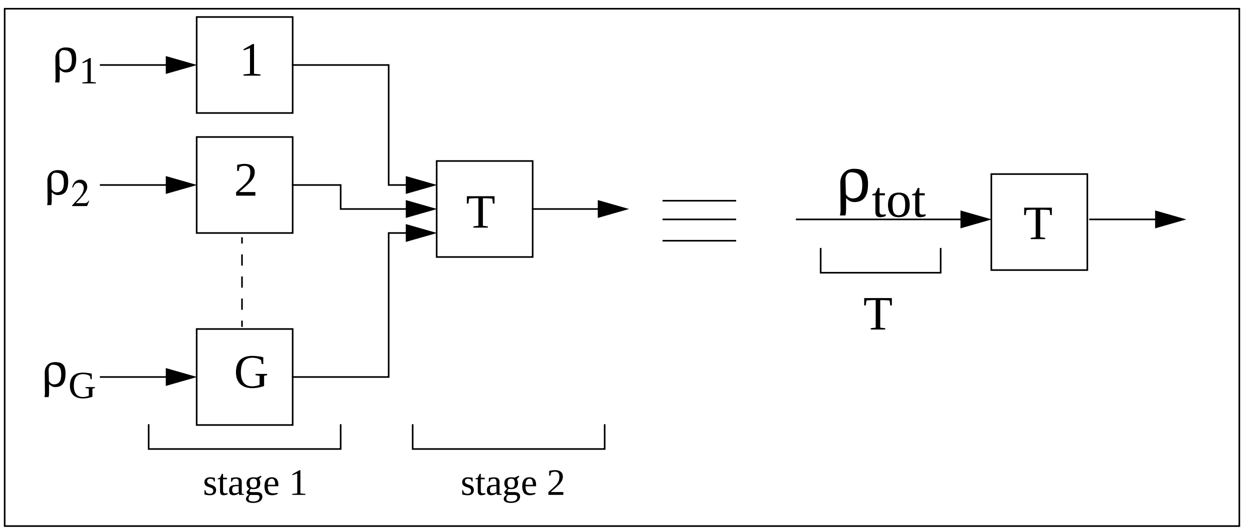


We thus can reduce analysis of multi-stage systems to analysis of 2-stage systems with a superposition of the exogenous inputs.

Fun Example: Expected Occupancies...

Definition: The function #(ρ) represents the expected number of packets in a single stage system with loading ρ.



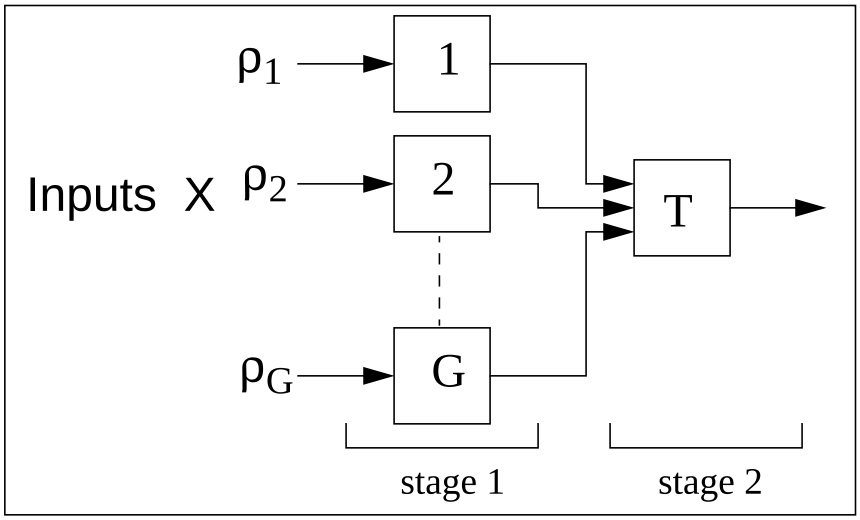


E(Number in Second Stage)

= E(Number in Total System) - E(Number in First Stage)

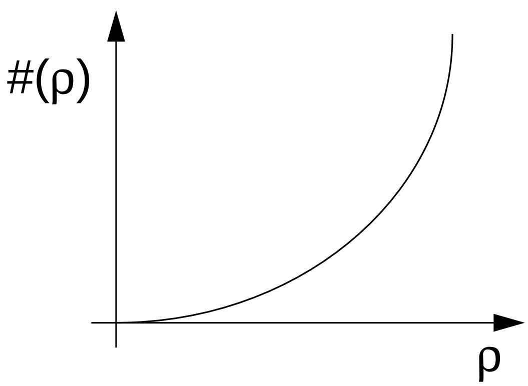
$= \rho_{tot} + \#(\rho_{tot}) \quad - \quad [\#(\rho_1) + ... + \#(\rho_G)].$

Convexity/Concavity of Expected Occupancy functions:

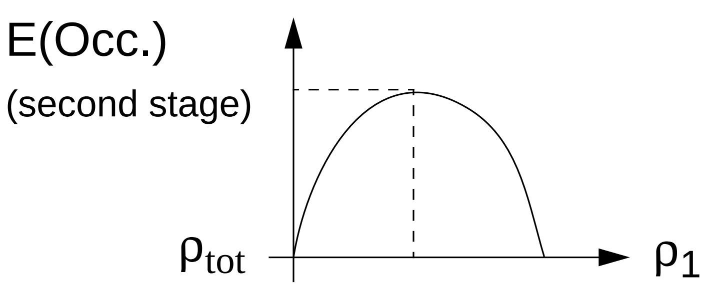


For distributable inputs X:

1) The #(ρ) function is <u>convex</u> and <u>monotically increasing</u>:

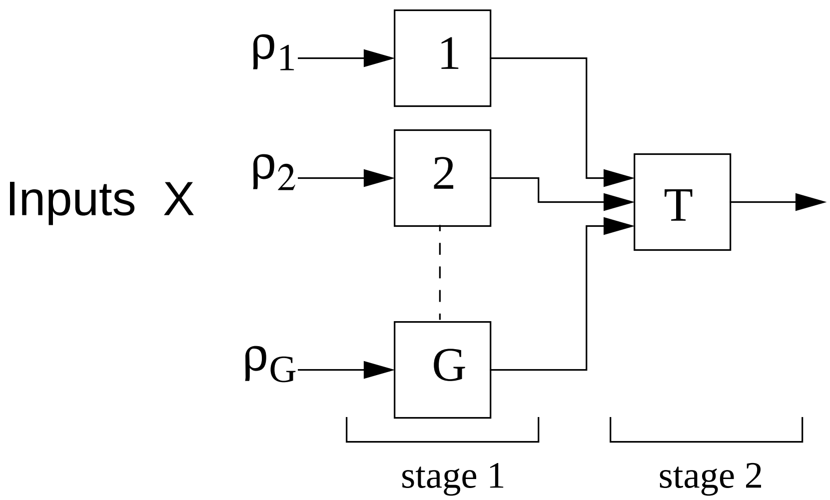


2) The expected second stage occupancy is a concave symmetric function in the input loadings (ρ_1, ρ_2, ..., ρ_G).



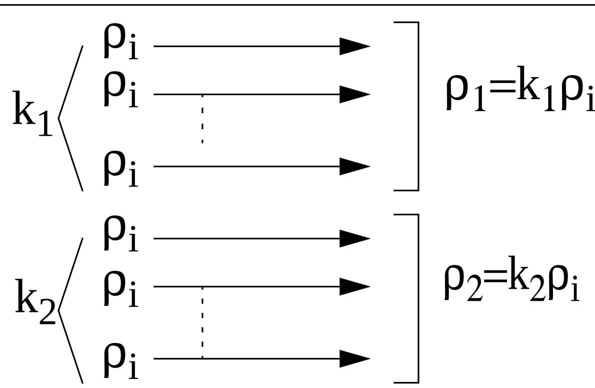
Thus, Expected Occupancy in the Second Stage is Maximized at the Uniform Distribution.

Interestingly, this is also the Maximum Entropy Solution.

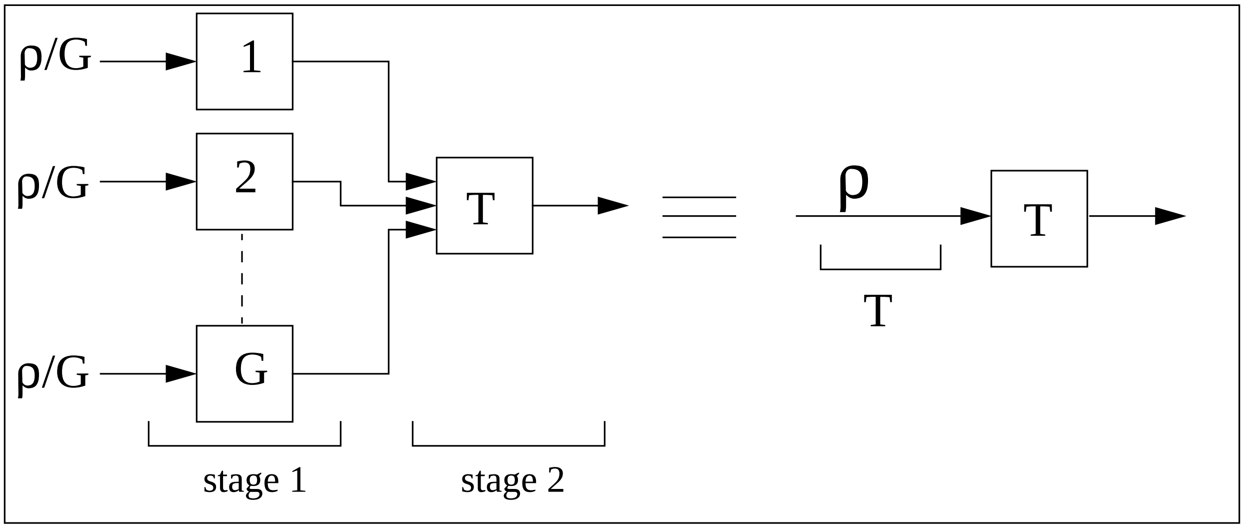


<u>Definition</u> : A "distributable input" X is an input which can be written as the sum of iid component processes.

$$X = \sum_{k=1}^{M} x_i .$$



Example: Expected Occupancy in second stage:



E(Number in Second Stage)

= E(Number in Total System) - E(Number in First Stage)

= ρ + #(ρ) - G#(ρ/G).

Ex: Memoryless Input => $\#(\rho) = \rho + \frac{\rho^2}{2(1-\rho)}$. (M/D/1)

=> E(Number in second stage) =

$$\rho + \#(\rho) - G\left[\frac{\rho}{G} + \frac{(\rho/G)^2}{2(1-\rho/G)}\right] \longrightarrow \#(\rho)$$

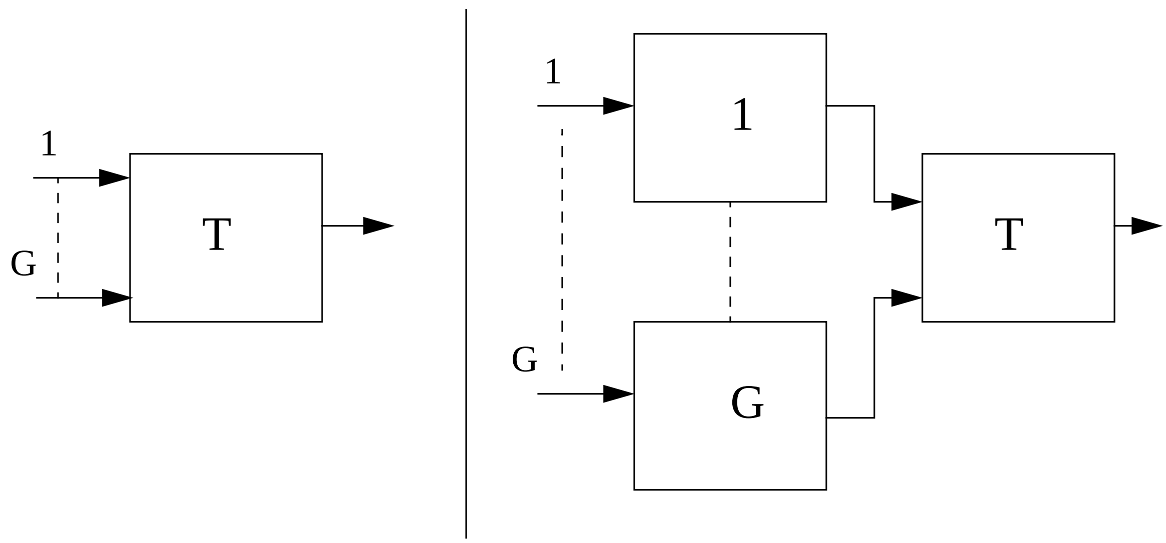
(Thus, second stage looks like M/D/1 system as G gets large).

Motivation:

We want to multiplex many users together and serve them one at a time.
Suppose that our single stage muxer does not physically have enough input ports to accomodate all of our users. “Pre-stages” need to be installed.

Q: How does the 2-stage system compare to the original 1-stage system?

Muli-Stage Multiplexing Conjecture:



Compare the two systems with identical inputs 1,...,G. Suppose we want to preserve a sufficiently low packet drop probability ε.

1) The total number of buffer slots needed in the 2-stage system will be no less than the number needed in the 1-stage system. (Hence: multi-staging is “sub-optimal”).

2) The total number of buffer slots needed in the final stage of the 2-stage system will be no more than the number needed in the 1-stage system.

Description of Observation Window (Same as a G/D/infinity queue).

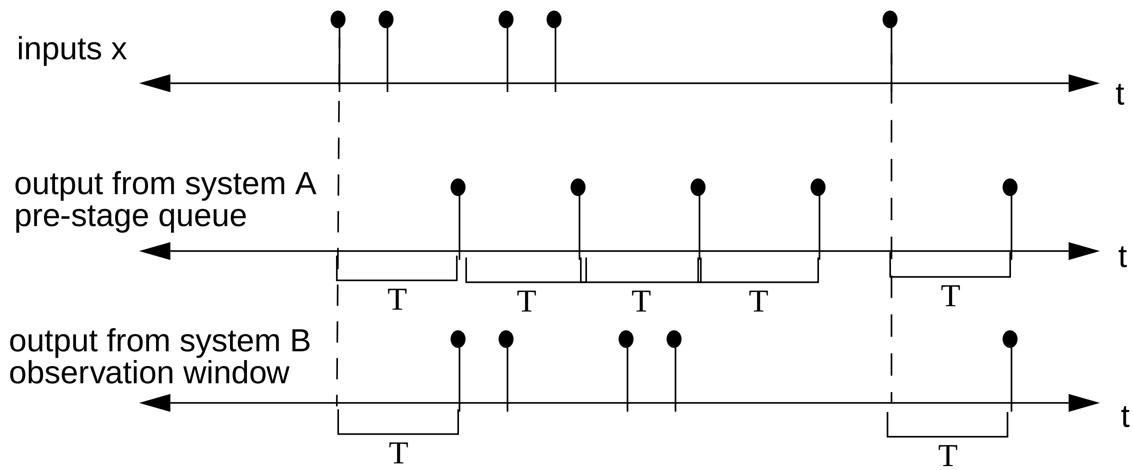


Fig. 2.12: An example timing diagram of inputs x and the corresponding outputs in both the system A pre-stage queue and the system B observation window.