Transfer Lines and Decomposition Techniques

Stanley B. Gershwin
gershwin@mit.edu,http://web.mit.edu/manuf-sys

April 16, 2008

Politecnico di Milano

Overview

- Basic issues — manufacturing systems fundamentals
- Basic issues — mathematics fundamentals
- Single-machine-factory behavior
- Factories with infinite buffers
- Factories without buffers
- Two-machine lines with buffers
- Long lines with buffers
- HP case
- Conclusion

Basic Issues

- Frequent new product introductions.
- Product lifetimes often short.
- Process lifetimes often short.

This leads to short factory lifetimes and frequent building and rebuilding of factories.

There is little time for improving the factory after it is built; it must be built right.

Basic Issues

Consequent Needs

- Tools to predict performance of proposed factory design.
- Tools for optimal real-time management (control) of factories.
- Manufacturing Systems Engineering professionals who understand factories as complex systems.

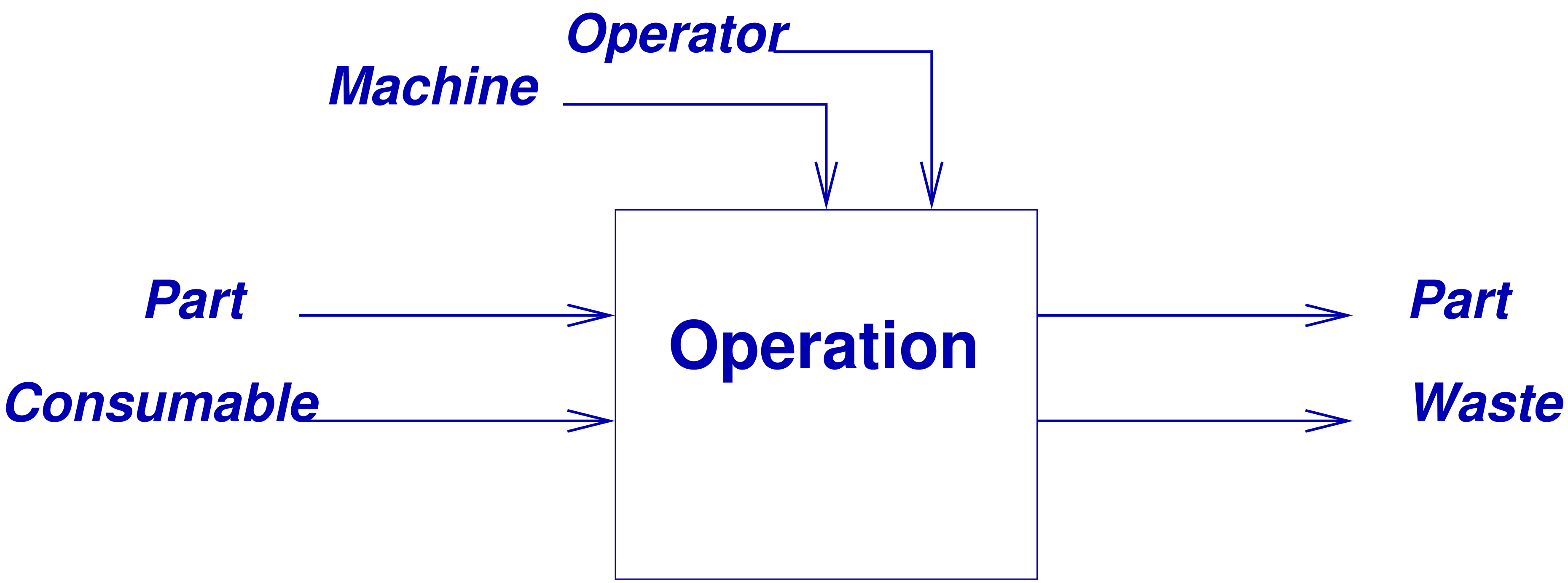
Basic Issues

- Quantity – how much and when.
- Quality – how well.

General Statement: Variability is the enemy of manufacturing.

Basic Issues

Operation



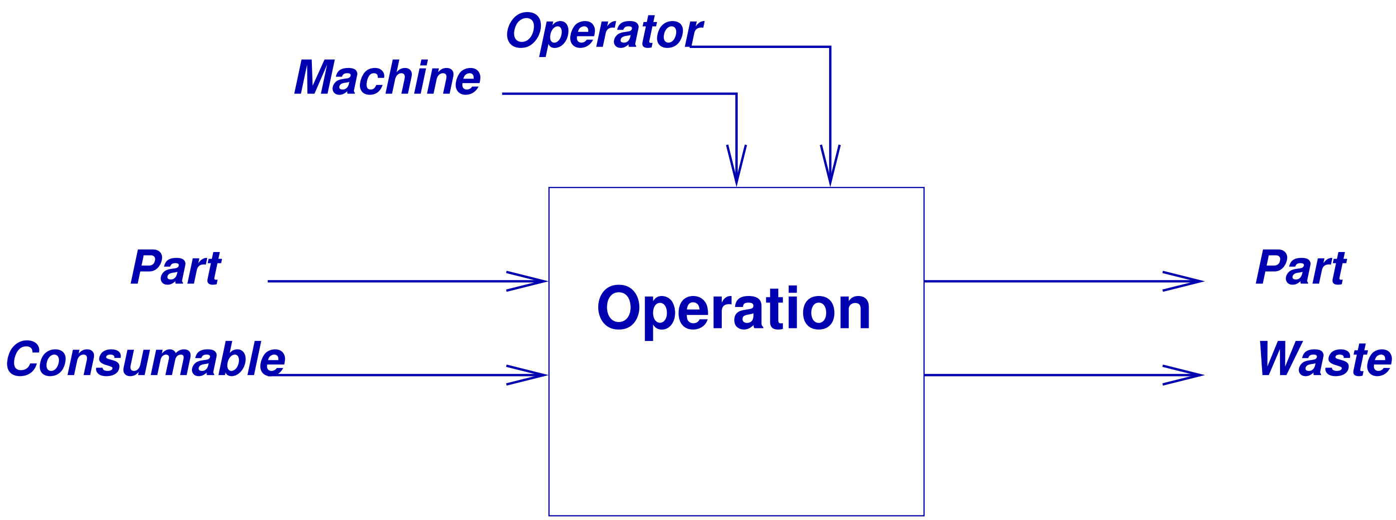
Nothing happens until everything is present.

Basic Issues

Whatever does not arrive last must wait.

- *Inventory:* parts waiting.
- *Under-utilization:* machines waiting.
- *Idle work force:* operators waiting.

Basic Issues



- *Reductions* in the availability, or ...
- *Variability* in the availability ...

... of any one of these items causes waiting in the rest of them and reduces performance of the system.

Basic Issues

- *Uncertainty:* Incomplete knowledge.
- *Variability:* Change over time.
- *Randomness:* A specific kind of incomplete knowledge that can be quantified and for which there is a mathematical theory.

Basic Issues

- Factories are full of random events:
  - ⋆ machine failures
  - ⋆ changes in orders
  - ⋆ quality failures
  - ⋆ human variability
- The economic environment is uncertain:
  - ⋆ demand variations
  - ⋆ supplier unreliability
  - ⋆ changes in costs and prices

Basic Issues

Therefore, factories should be

- *designed* and *operated*

to minimize the

- *creation, propagation,* or *amplification*

of uncertainty, variability, and randomness.

Basic Issues

- A *Markov process* is a stochastic process in which the probability of finding X at some value at time $t + \delta t$ depends only on the value of X at time t.

- Or, let $x(s), s \leq t$, be the history of the values of X before time t and let A be a possible value of X. Then

  prob$\{X(t+\delta t) = A | X(s) = x(s), s \leq t\}$ =
  prob$\{X(t+\delta t) = A | X(t) = x(t)\}$

Basic Issues

- In words: if we know what X was at time t, we don't gain any more useful information about $X(t + \delta t)$ by *also* knowing what X was at any time earlier than t.

Markov processes

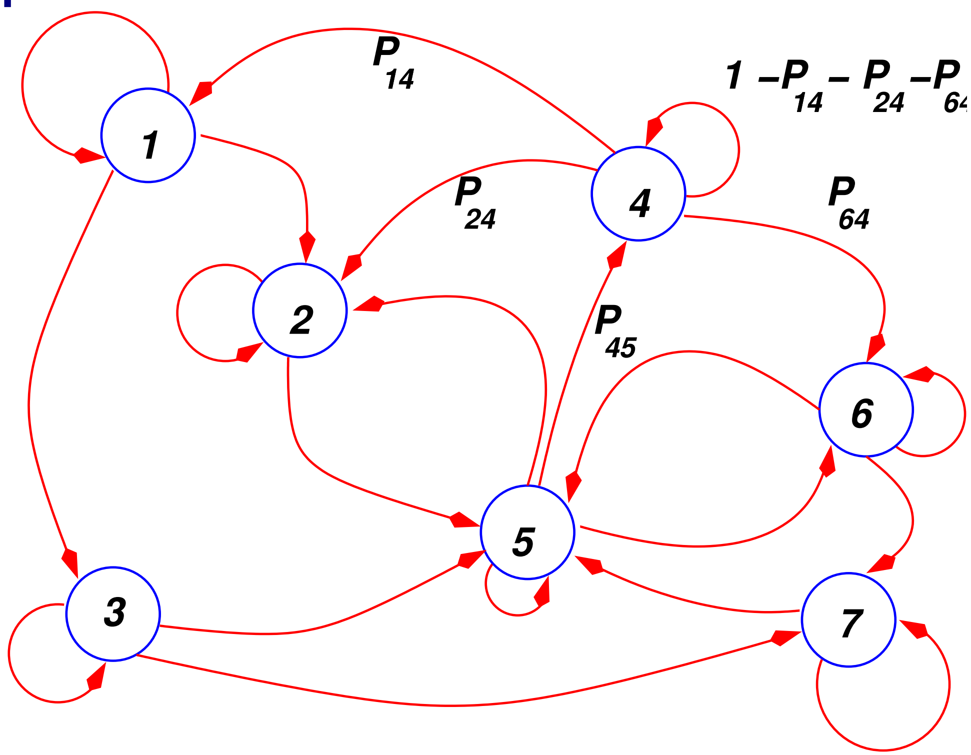
States and transitions

Discrete state, discrete time

- States can be numbered 0, 1, 2, 3, ... *(or with multiple indices if that is more convenient)*.
- Time can be numbered 0, 1, 2, 3, ... *(or 0, Δ, 2Δ, 3Δ, ... if more convenient)*.
- The probability of a transition from j to i in one time unit is often written P_{ij}, where

$$P_{ij} = \text{prob}\{X(t+1) = i | X(t) = j\}$$

Transition graph



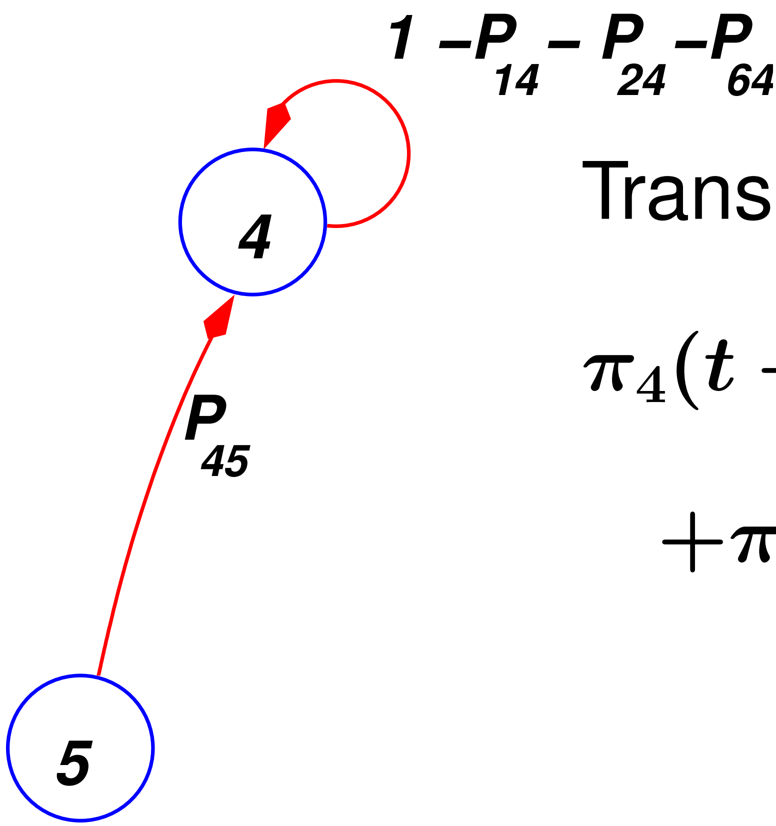
P_{ij} is a probability. Note that $P_{ii} = 1 - \sum_{m, m \neq i} P_{mi}$.

Markov processes

States and transitions

Discrete state, discrete time

- Define $\pi_i(t) = \text{prob}\{X(t) = i\}$.
- Transition equations: $\pi_i(t+1) = \sum_j P_{ij}\pi_j(t)$.
- Normalization equation: $\sum_i \pi_i(t) = 1$.



Transition equation:

$$\pi_4(t+1) = \pi_5(t)P_{45}$$

$$+\pi_4(t)(1 - P_{14} - P_{24} - P_{64})$$

Markov processes

States and transitions

Discrete state, discrete time

- *Steady state:* $\pi_i = \lim_{t \to \infty} \pi_i(t)$, if it exists.

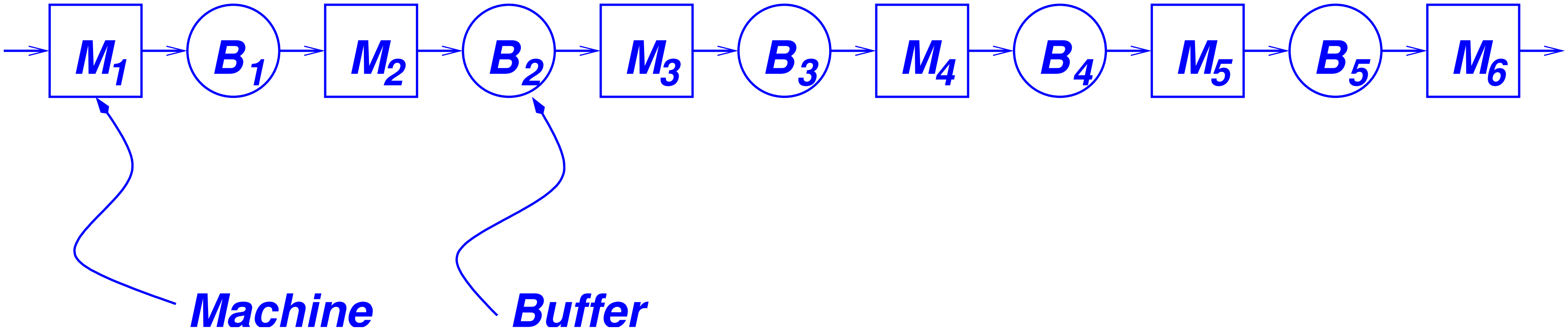
- Steady-state transition equations: $\pi_i = \sum_j P_{ij} \pi_j$.

- *Alternatively,* steady-state balance equations:
  $\pi_i \sum_{m, m \neq i} P_{mi} = \sum_{j, j \neq i} P_{ij} \pi_j$

- Normalization equation: $\sum_i \pi_i = 1$.

Flow Line

... also known as a Production or Transfer Line.



- Machines are unreliable.
- Buffers are finite.
- Ideal for high-volume, low variety production.

In the following, we assume that there is a single product.

Flow Line

Question: Why study flow lines? They are a very special case of manufacturing systems

Answer: Because they exhibit many of the important characteristics of any manufacturing system in a pure form.

They clarify the relationships among random variability and capacity, inventory, and quality.

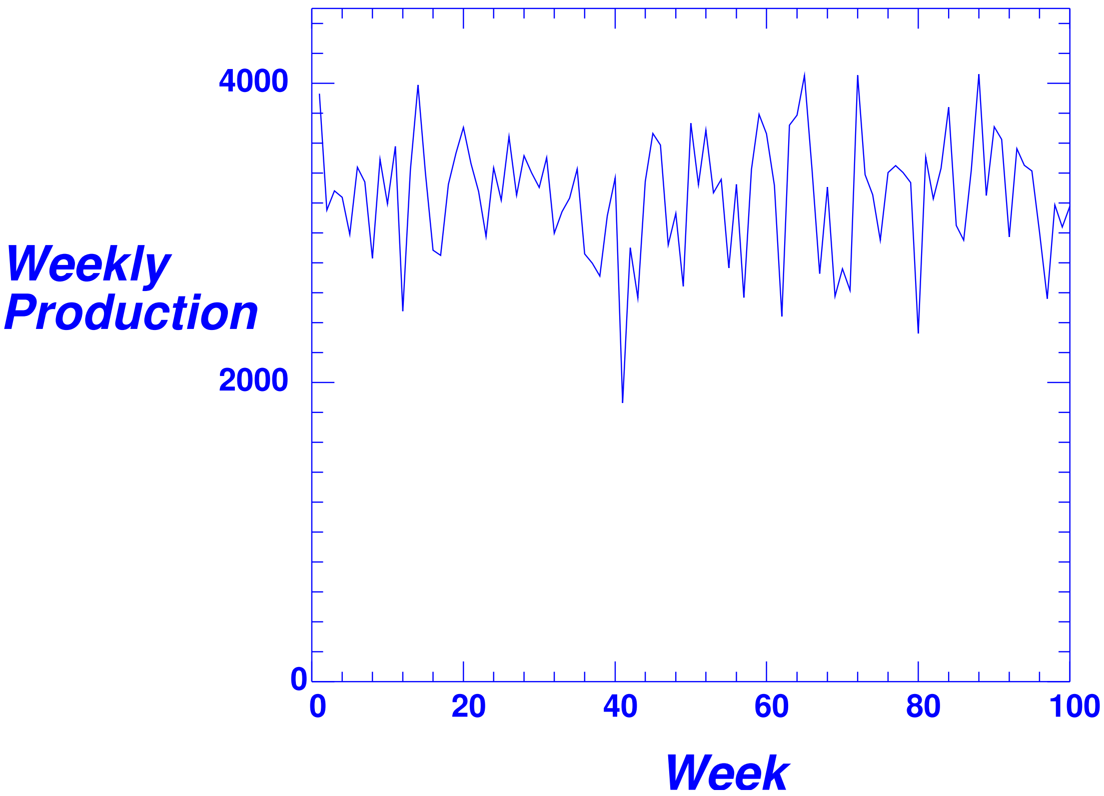
Also, they still account for substantial economic activity.

Flow Line

- Production rate
- Inventory
- Lead Time

Means are always important. Sometimes *standard deviations* and *percentiles* are also important.

Flow Line



Production output from a simulation of a transfer line.

Single Reliable Machine

If a machine is perfectly reliable, and its average operation time is τ, then its maximum production rate is

$$\frac{1}{\tau}$$

Single Unreliable Machine

Machine is either *up* or *down* .

- MTTF = mean time to fail.
- MTTR = mean time to repair
- MTBF = MTTF + MTTR

Production rate

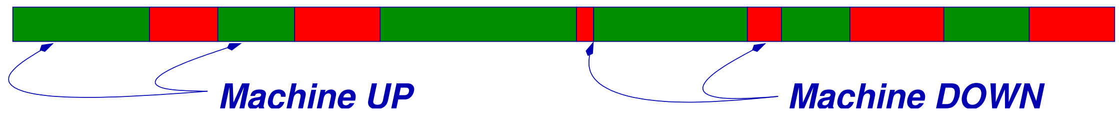
- If the machine is unreliable, and
  - ⋆ its average operation time is τ,
  - ⋆ its mean time to fail is MTTF,
  - ⋆ its mean time to repair is MTTR,

  then its maximum production rate is

$$\frac{1}{\tau}\left(\frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}\right) = \frac{1}{\tau}e$$

  where

$$e = \textit{efficiency} = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}$$



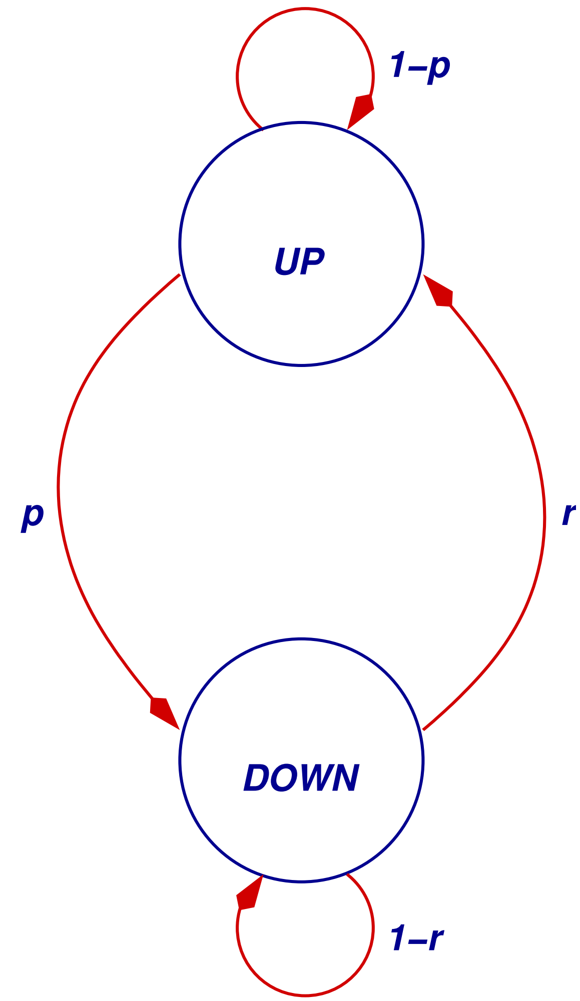
- Average production rate, while machine is up, is $1/\tau$.
- Average duration of an up period is MTTF.
- Average production during an up period is MTTF/τ.
- Average duration of up-down period: MTTF + MTTR.
- Average production during up-down period: MTTF/τ.
- Therefore, average production rate is (MTTF/τ)/(MTTF + MTTR).

If failure and repair times are geometrically distributed, then

$$p = \tau/\text{MTTF}$$

$$r = \tau/\text{MTTR}$$

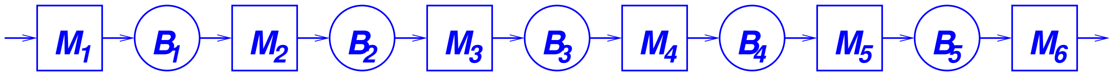
$$e = \frac{r}{r+p}$$



Production Rates

- So far, the machine really has *three* production rates:
  - $\star$ $1/\tau$ when it is up *(short-term capacity)* ,
  - $\star$ 0 when it is down *(short-term capacity)* ,
  - $\star$ $(1/\tau)(\text{MTTF}/\text{MTBF})$ on the average *(long-term capacity)* .

Infinite-Buffer Line



- Starvation: Machine M_i is starved at time t if Buffer B_{i-1} is empty at time t.

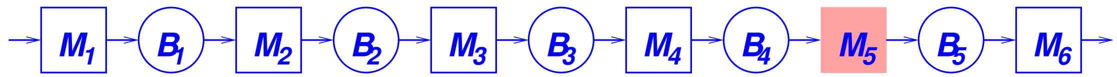
Assumptions:

- A machine is not idle if it is not starved.
- The first machine is never starved.

Infinite-Buffer Line

- Operation-Dependent Failures
  - ⋆ A machine can only fail while it is working — not idle.
  - ⋆ *(When buffers are finite, idleness also occurs due to blockage.)*
  - ⋆ *IMPORTANT!* MTTF *must* be measured in working time!
  - ⋆ This is the usual assumption.

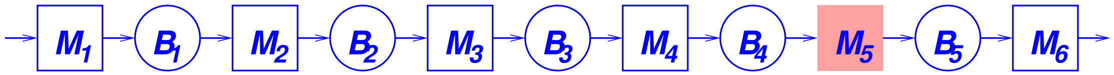
Infinite-Buffer Line



- The production rate of the line is the production rate of the *slowest* machine in the line — called the *bottleneck* .
- Slowest means *least average production rate* , where average production rate is given by

$$\frac{1}{\tau_i} e_i$$

Infinite-Buffer Line

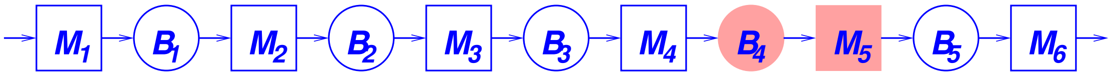


- Production rate is therefore

$$P = \min_i \frac{1}{\tau_i} \left(\frac{\text{MTTF}_i}{\text{MTTF}_i + \text{MTTR}_i} \right)$$

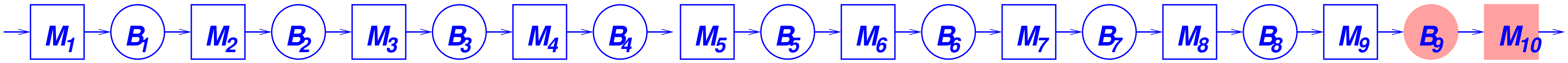
- and M_i is the bottleneck.

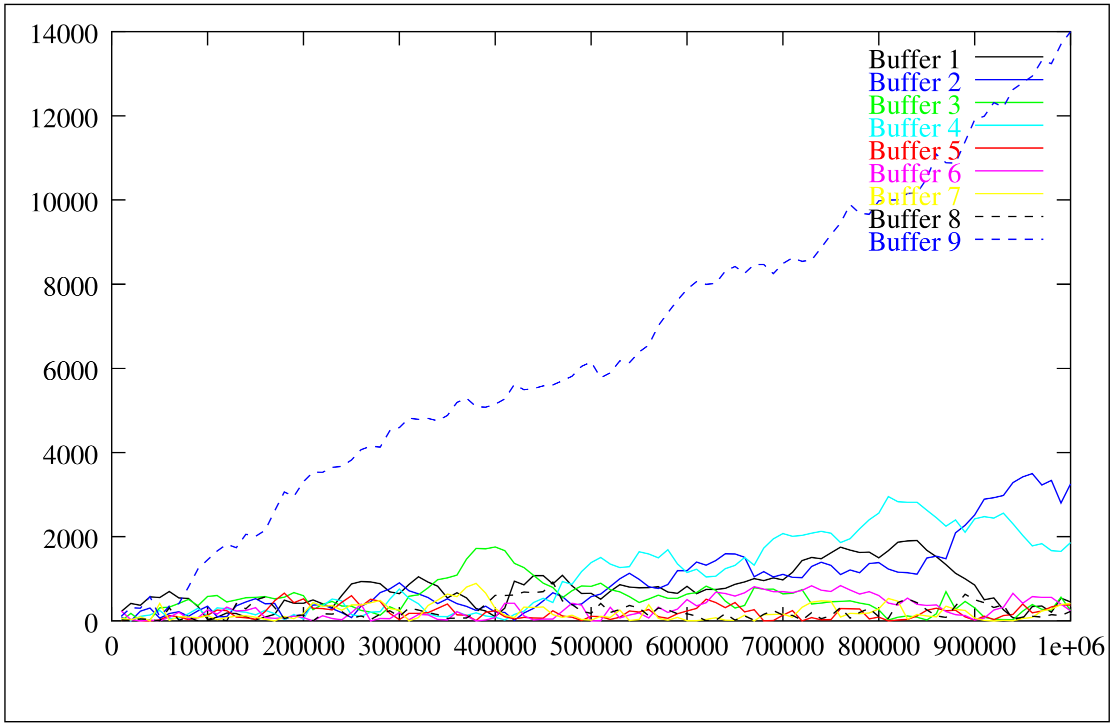
Infinite-Buffer Line



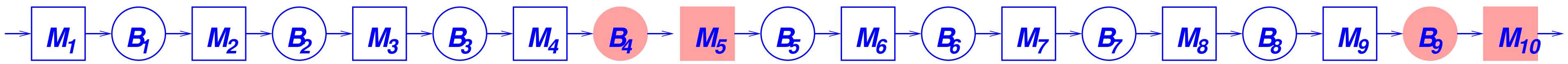
- The system is not in steady state.
- Inventory accumulates without limit in the buffer upstream of the bottleneck.
- A finite amount of inventory appears downstream of the bottleneck.

Infinite-Buffer Line



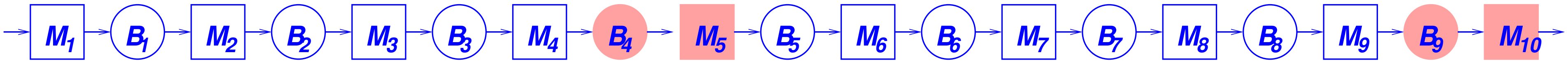


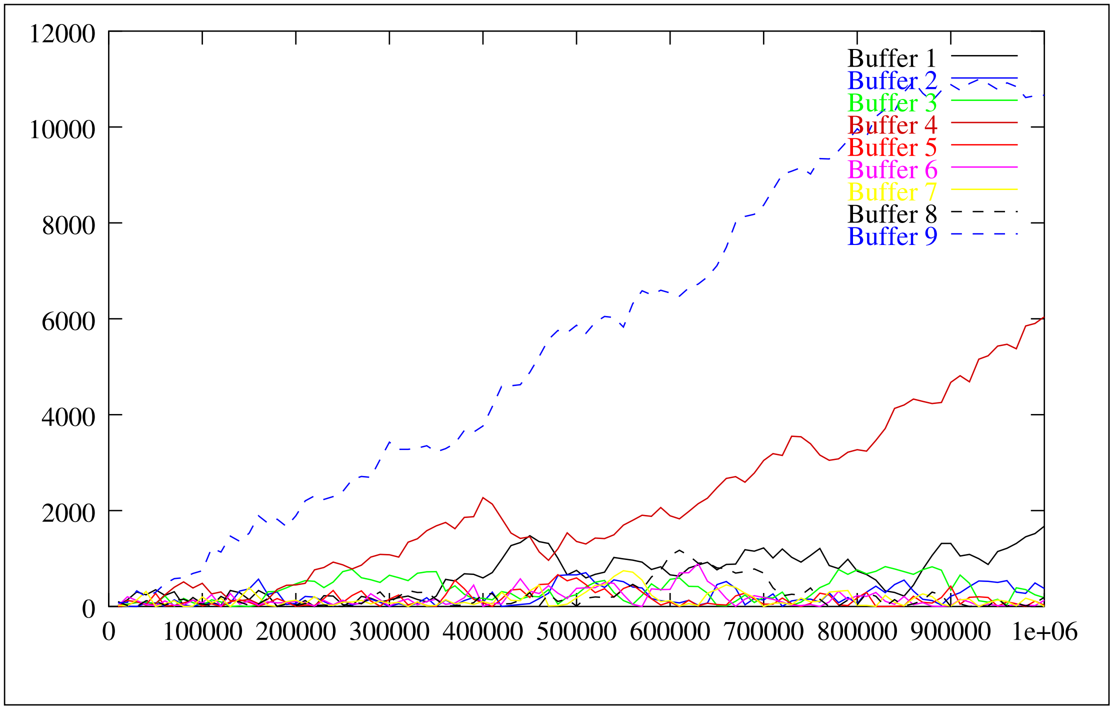
Infinite-Buffer Line



- The *second bottleneck* is the slowest machine upstream of the bottleneck. An infinite amount of inventory accumulates just upstream of it.
- A finite amount of inventory appears between the second bottleneck and the machine upstream of the first bottleneck.
- Et cetera.

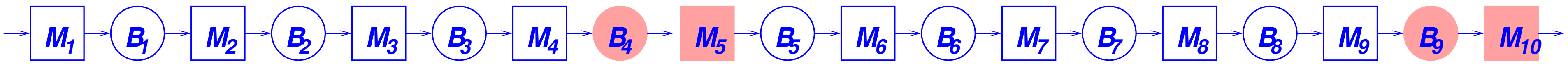
Infinite-Buffer Line

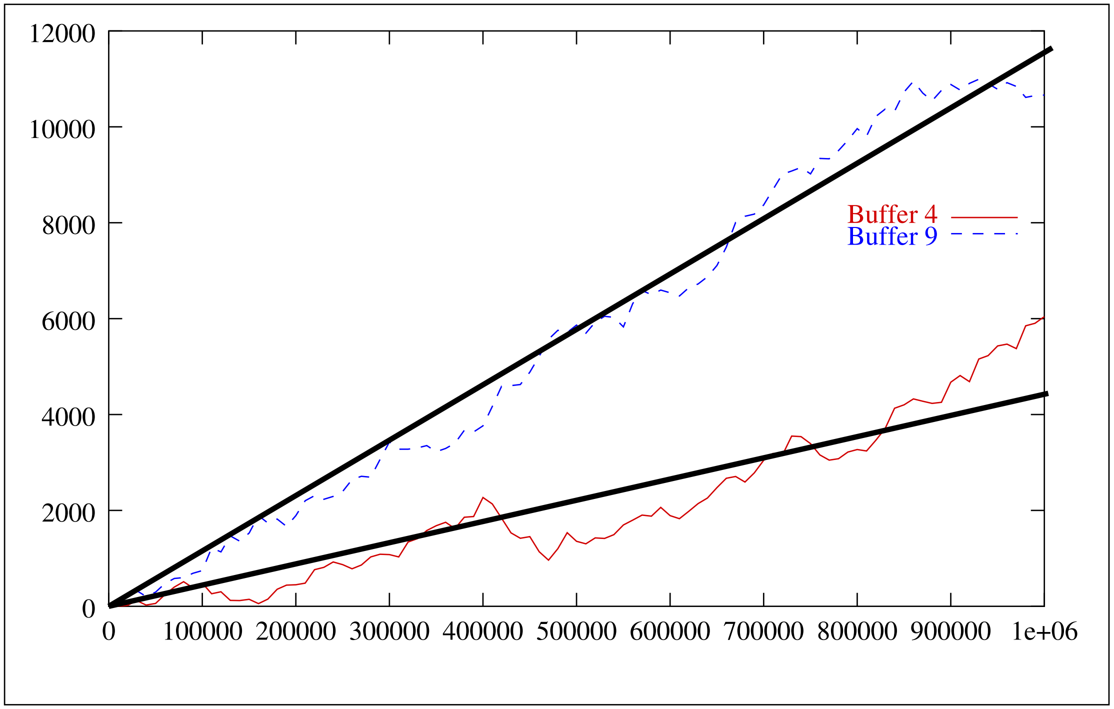




A 10-machine line with bottlenecks at Machines 5 and 10.

Infinite-Buffer Line





Question:

- What are the slopes *(roughly!)* of the two indicated graphs?

Infinite-Buffer Line

Questions:

- If we want to increase production rate, which machine should we improve?
- What would happen to production rate if we improved any other machine?

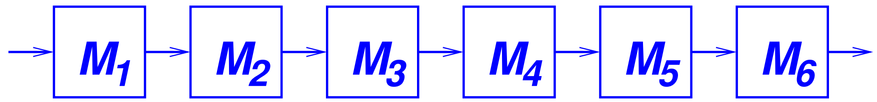
Infinite-Buffer Line

Discussion

- If the buffers were finite, how would that affect the behavior and performance of the system?

- Is randomness the main issue here, or can it be summarized by an average?

Zero-Buffer Line

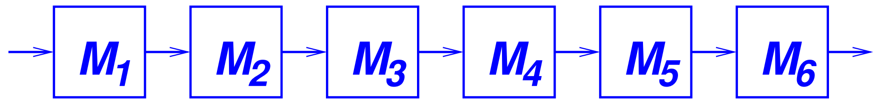


- If any one machine fails, or takes a very long time to do an operation, *all* the other machines must wait.

- Therefore the production rate is usually less — usually much less – than that of the slowest machine.

Zero-Buffer Line

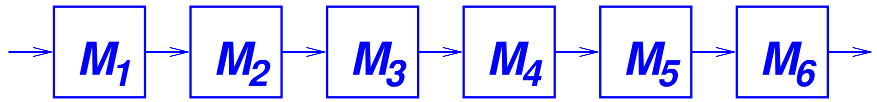
Constant, equal operation times, unreliable machines



- *Assumption:* Failure and repair times are *geometrically* distributed.
- Define $p_i = \tau/\text{MTTF}_i$ = probability of failure during an operation.
- Define $r_i = \tau/\text{MTTR}_i$ probability of repair during an interval of length τ when the machine is down.

Zero-Buffer Line

Constant,
equal operation times,
unreliable machines



Buzacott's Zero-Buffer Line Formula:

Let k be the number of machines in the line. Then

$$P = \frac{1}{\tau} \frac{1}{1 + \sum_{i=1}^{k} \frac{p_i}{r_i}}$$

Zero-Buffer Line

Constant, equal operation times, unreliable machines

- Note that P is a function of the *ratio* p_i/r_i and not p_i or r_i separately.
- The same statement is true for the infinite-buffer line.
- However, the same statement is *not* true for a line with finite, non-zero buffers.

Zero-Buffer Line

Discussion

Questions:

- If we want to increase production rate, which machine should we improve?
- What would happen to production rate if we improved any other machine?

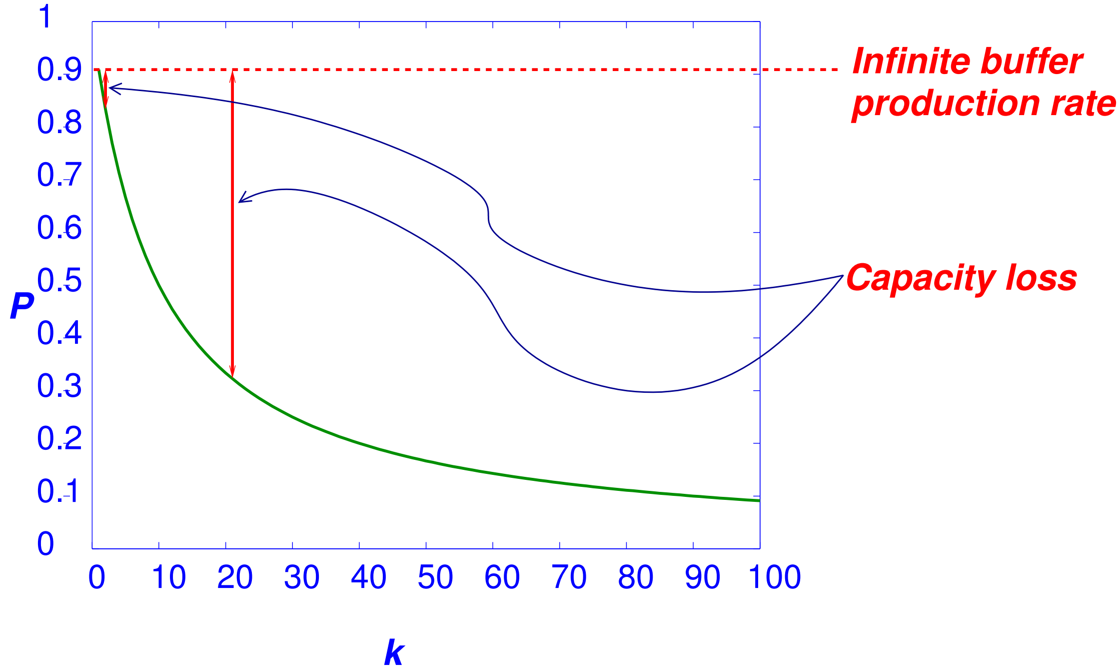
Zero-Buffer Line

- If the buffers were non-zero, how would that affect the behavior and performance of the system?
- Is randomness the main issue here, or can it be summarized by an average?

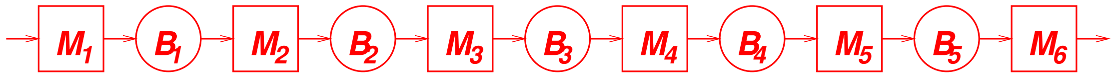
Zero-Buffer Line

P as a function of k

All machines are the same. As the line gets longer, the production rate decreases.

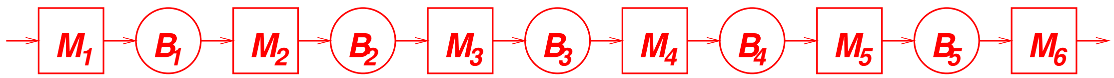


Finite-Buffer Lines



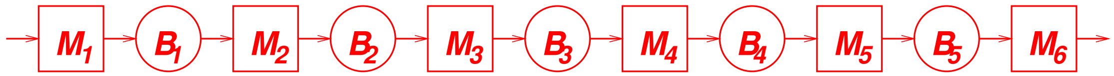
- Motivation for buffers: recapture some of the lost production rate.
- Cost
  - ⋆ in-process inventory/lead time
  - ⋆ floor space
  - ⋆ material handling mechanism

Finite-Buffer Lines



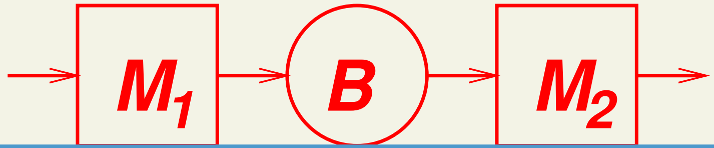
- **Infinite buffers:** delayed downstream propagation of disruptions(*starvation*).
- **Zero buffers:** instantaneous propagation in both directions.
- **Finite buffers:** delayed propagation in both directions.
  - ⋆ New phenomenon: *blockage* .
- **Blockage:** Machine M_i is blocked at time t if Buffer B_i is full at time t.

Finite-Buffer Lines

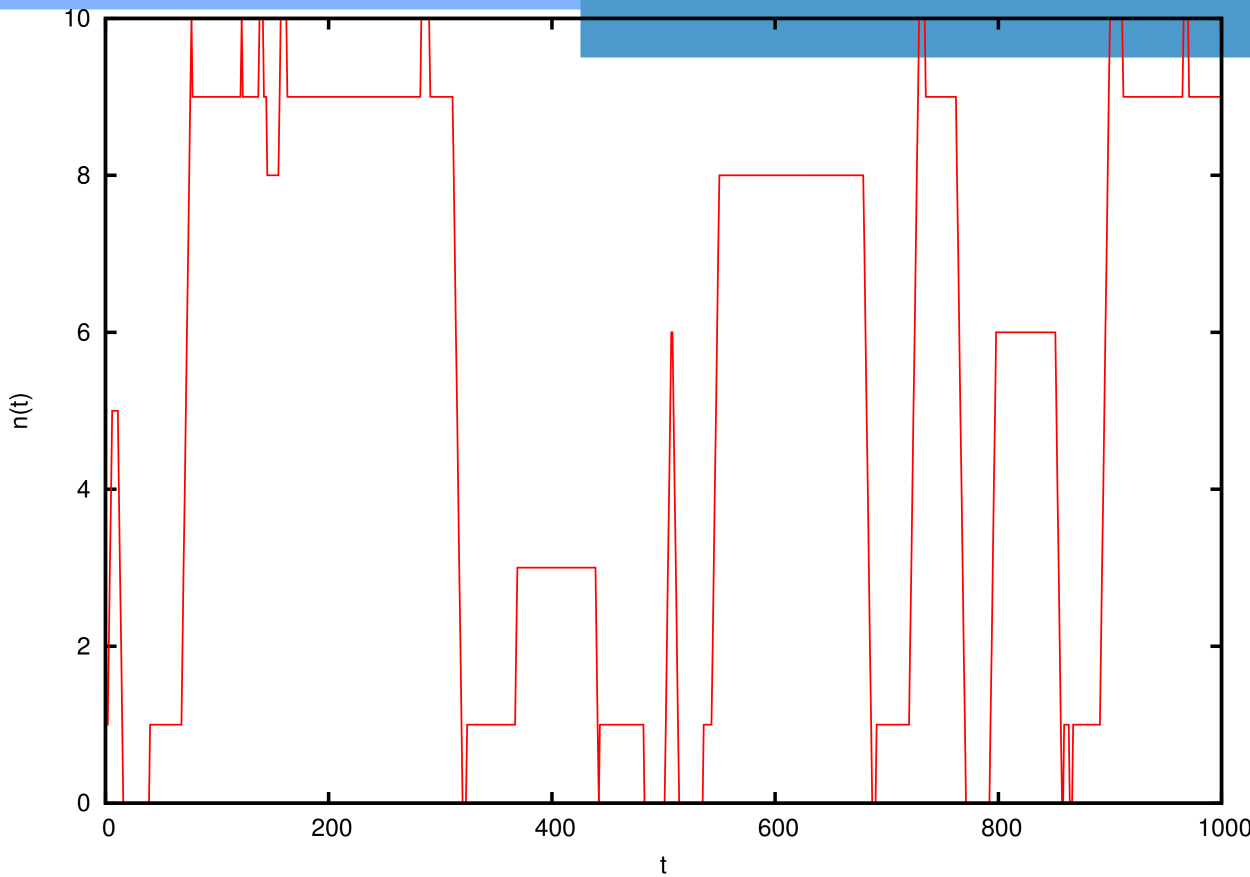


- Difficulty:
  - ⋆ No simple formula for calculating production rate or inventory levels.
- Solution:
  - ⋆ Simulation
  - ⋆ Analytical approximation
  - ⋆ *Exact analytical solution for two-machine lines only.*

Finite-Buffer Lines

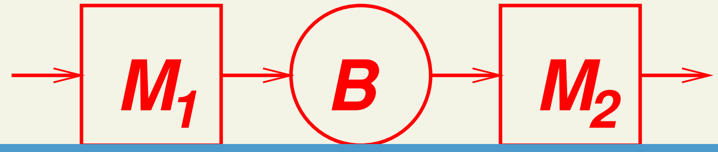


Simulations: Buffer Level vs. t

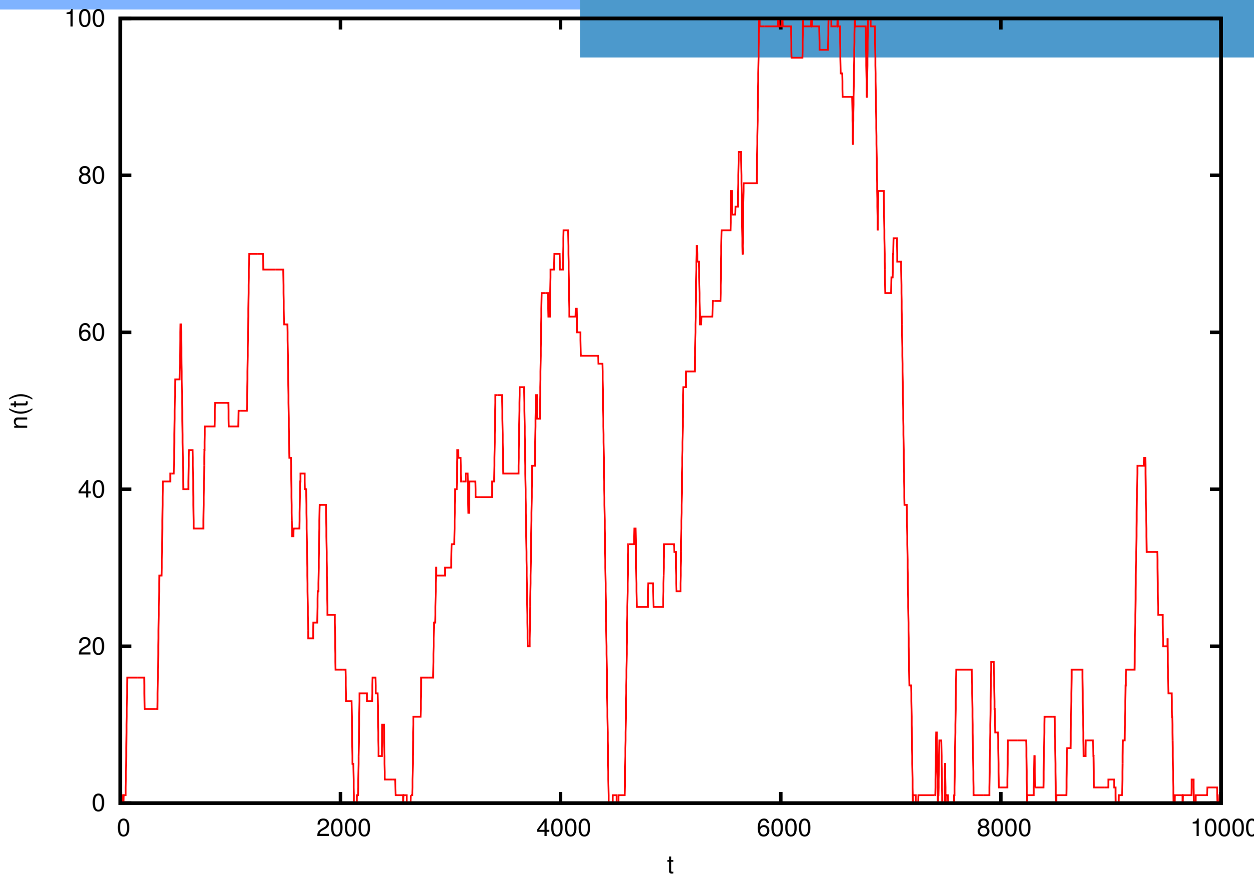


$$\text{MTTR}_i = 10, \text{MTTF}_i = 100, i = 1, 2; N = 10$$

Finite-Buffer Lines

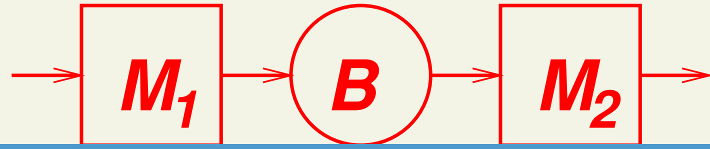


Simulations: Buffer Level vs. t

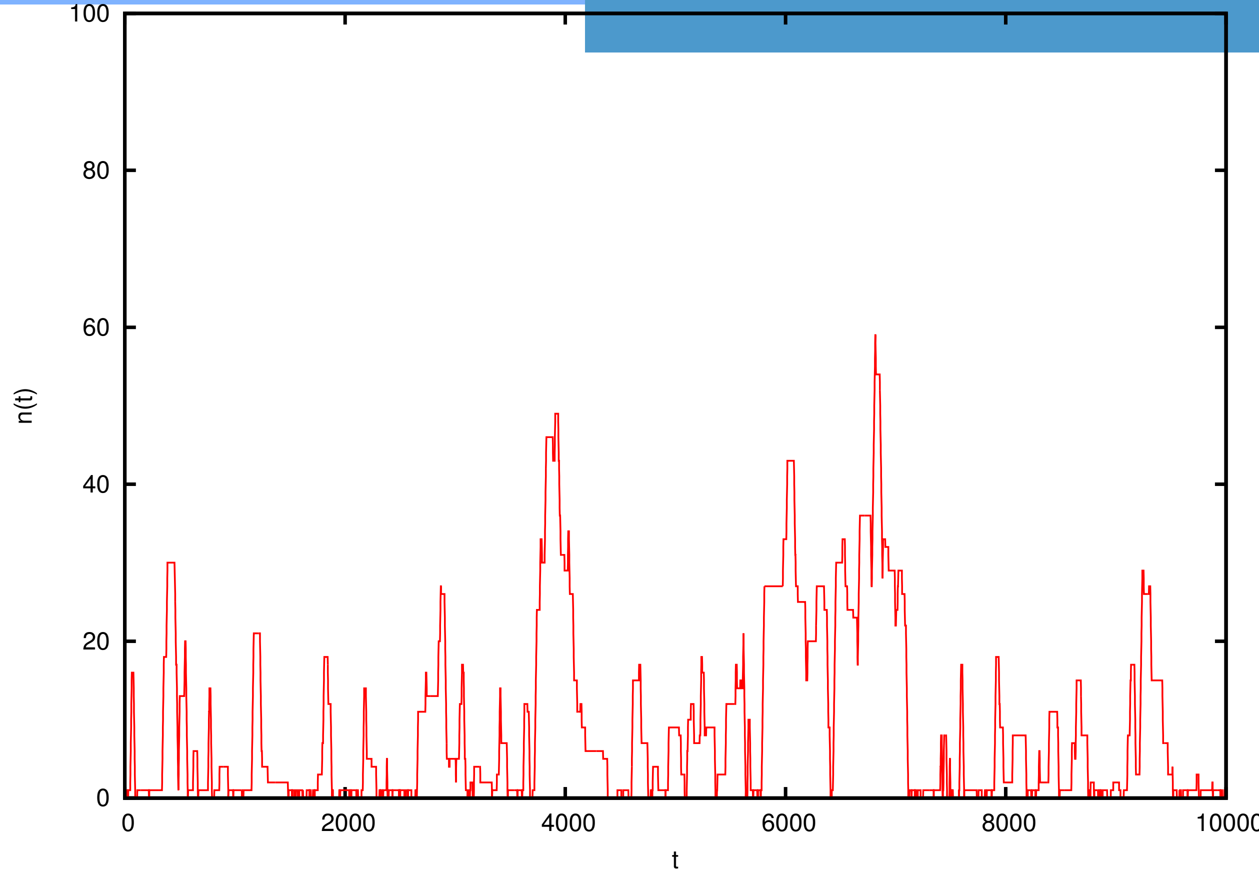


$$\text{MTTR}_i = 10, \text{MTTF}_i = 100, i = 1, 2; N = 100$$

Finite-Buffer Lines

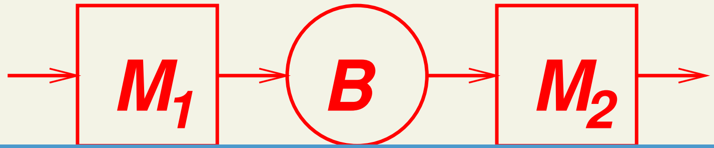


Simulations: Buffer Level vs. t

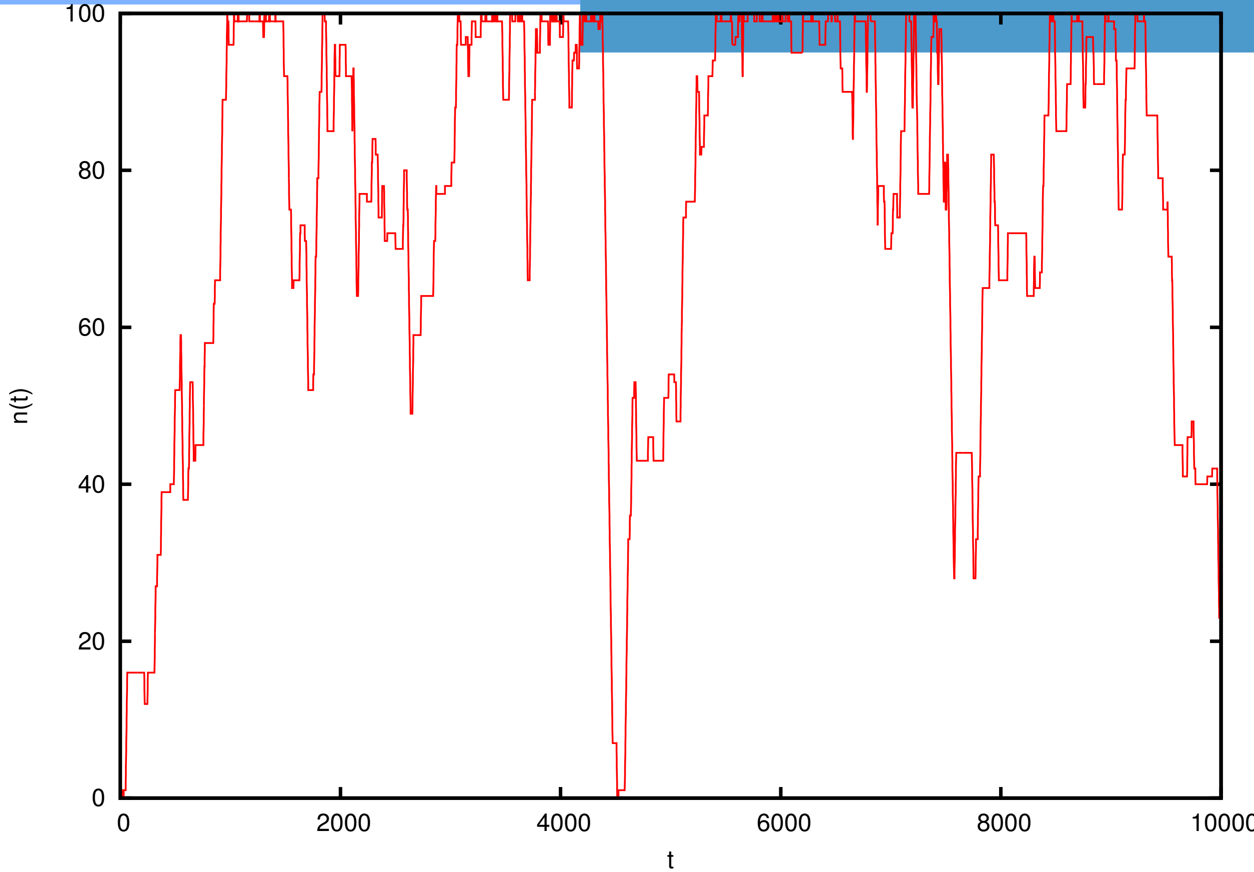


$$\text{MTTR}_i = 10, i = 1, 2, \text{MTTF}_1 = 50, \text{MTTF}_2 = 100; N = 100$$

Finite-Buffer Lines

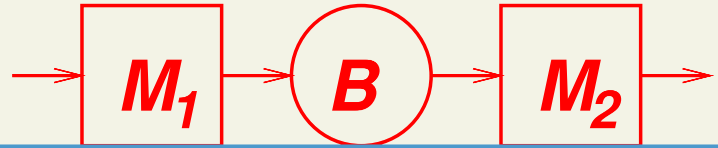


Simulations: Buffer Level vs. t



$$\text{MTTR}_i = 10, i = 1, 2, \text{MTTF}_1 = 100, \text{MTTF}_2 = 50; N = 100$$

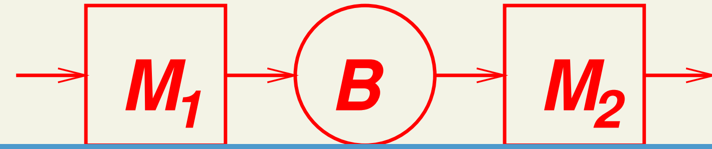
Finite-Buffer Lines



Simulations: Buffer Level vs. t

- How and why do the simulations differ?
- What are the consequences of the differences on the system performance?

Finite-Buffer Lines



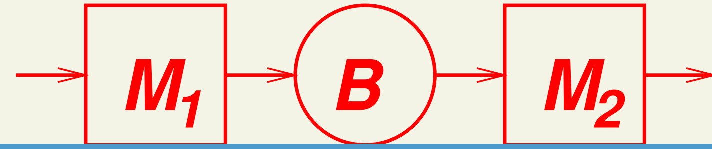
Model

- Exact solution *is* available to Markov process model.
- *Discrete time-discrete state Markov process:*

$$\text{prob}\{X(t+1)=x(t+1) | X(t)=x(t), X(t-1)=x(t-1), X(t-2)=x(t-2), \ldots\} =$$

$$\text{prob}\{X(t+1)=x(t+1) | X(t)=x(t)\}$$

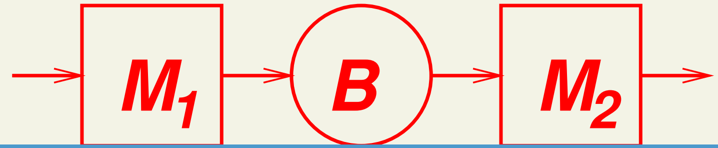
Finite-Buffer Lines



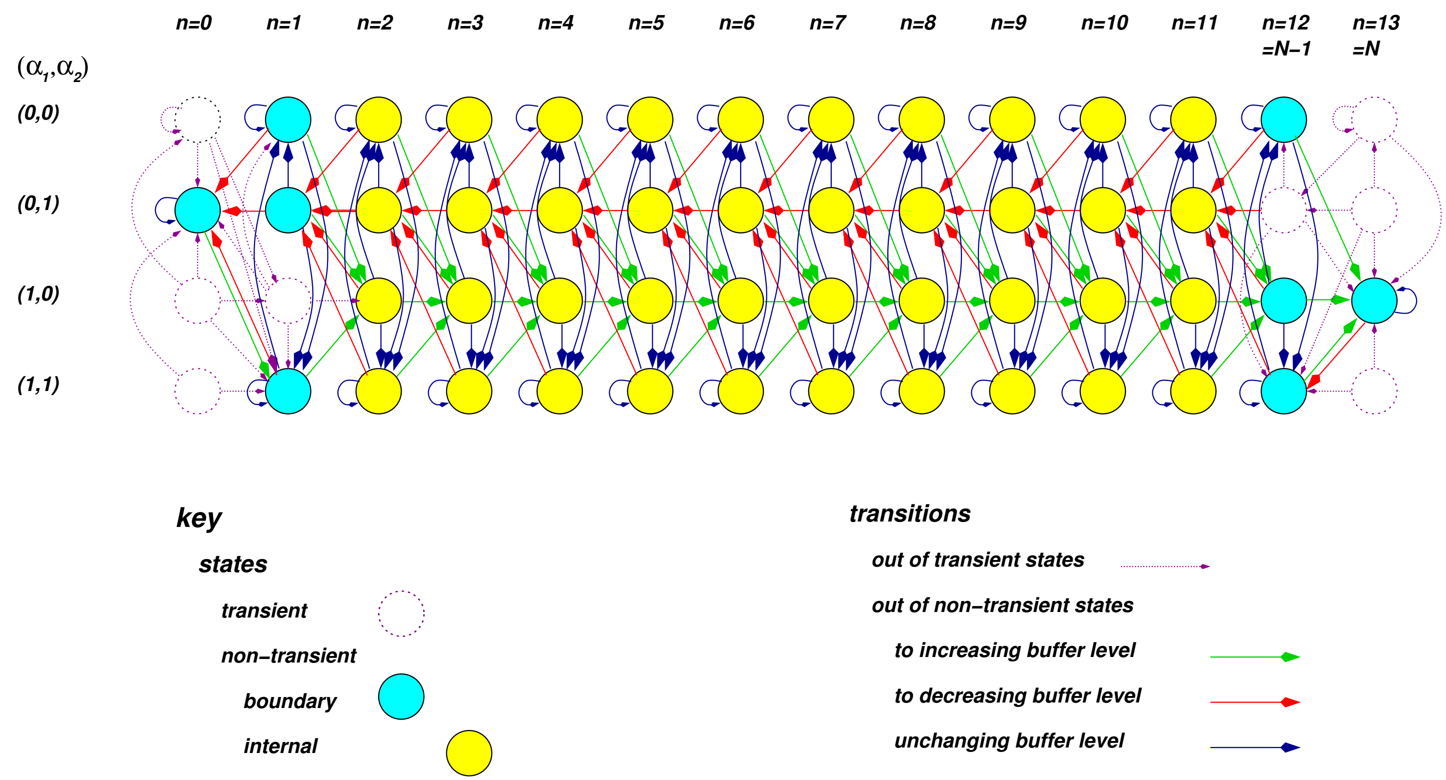
Model

Here, $X(t) = (n(t), \alpha_1(t), \alpha_2(t))$, where

- n is the number of parts in the buffer; $n = 0, 1, ..., N$.
- α_i is the repair state of M_i; $i = 1, 2$.
  - $\star$ $\alpha_i = 1$ means the machine is *up* or *operational*;
  - $\star$ $\alpha_i = 0$ means the machine is *down* or *under repair*.



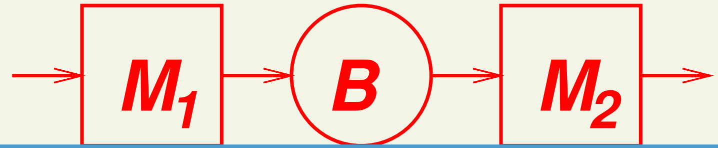
State Transition Graph for Deterministic Processing Time, Two–Machine Line



This is the model that was used to generate the graphs on the following slides.

Finite-Buffer Lines

Model



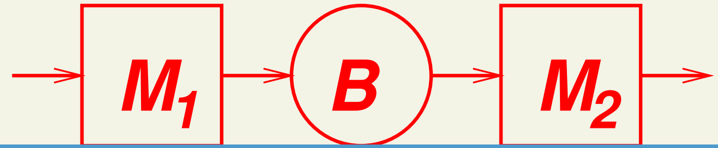
$$\text{production rate} = P = \sum_{n=1}^{N} \sum_{\alpha_1} \pi(n, \alpha_1, 1)$$

$$= \sum_{n=0}^{N-1} \sum_{\alpha_2} \pi(n, 1, \alpha_2)$$

Also, it can be shown that

$$P = \frac{r_1}{r_1 + p_1}(1 - \text{prob}(n = N)) = \frac{r_2}{r_2 + p_2}(1 - \text{prob}(n = 0))$$

Finite-Buffer Lines

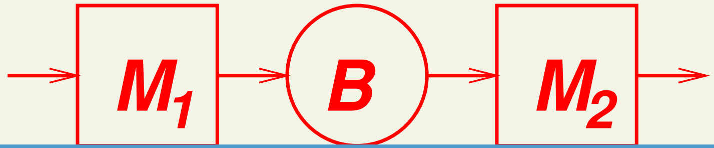


Model

Also

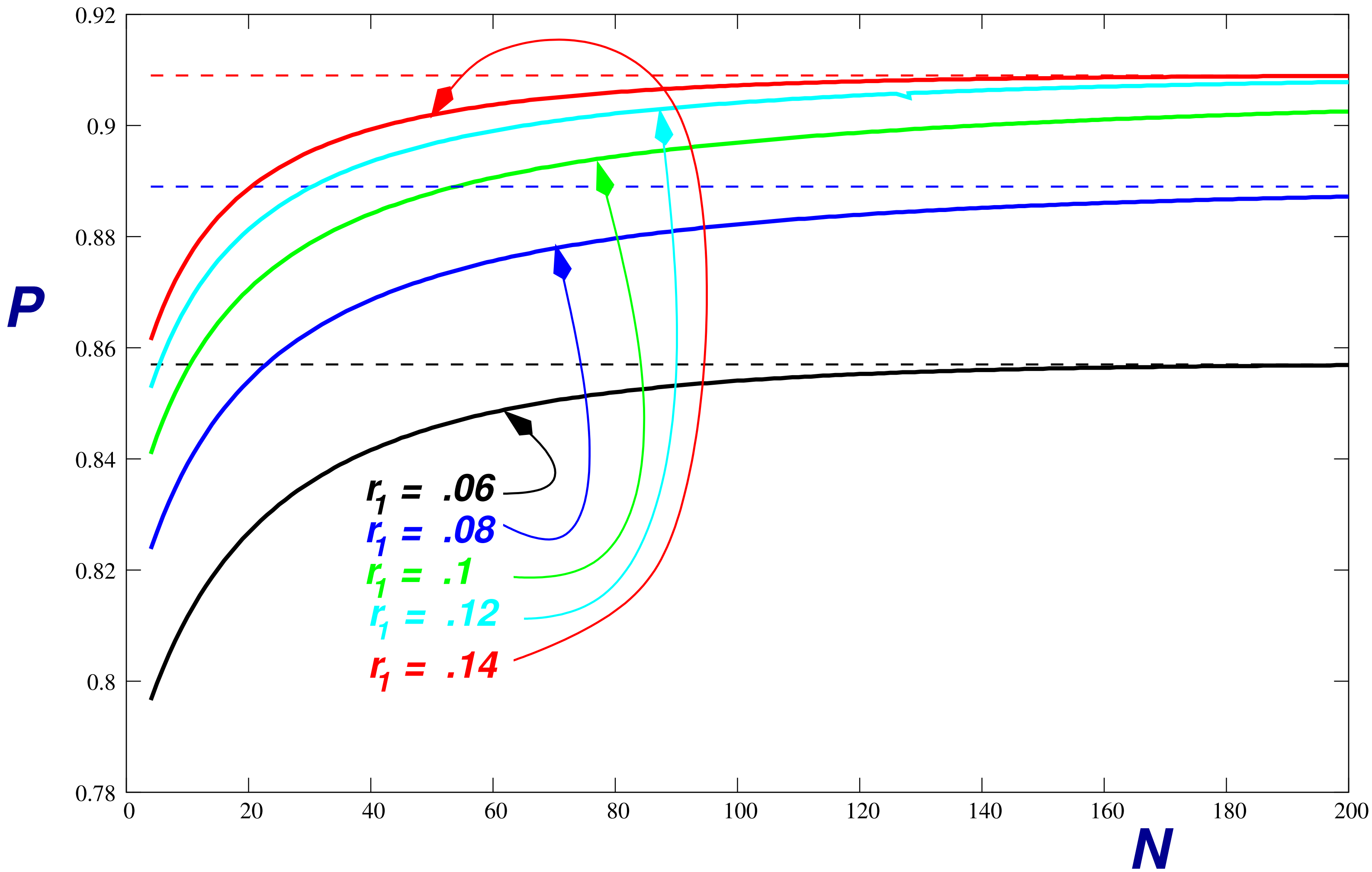
$$\text{average inventory} = \bar{n} = \sum_{n=1}^{N} \sum_{\alpha_1} \sum_{\alpha_2} n\pi(n, \alpha_1, \alpha_2)$$

Finite-Buffer Lines

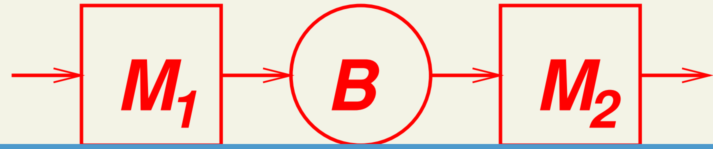


Production rate vs. Buffer Size

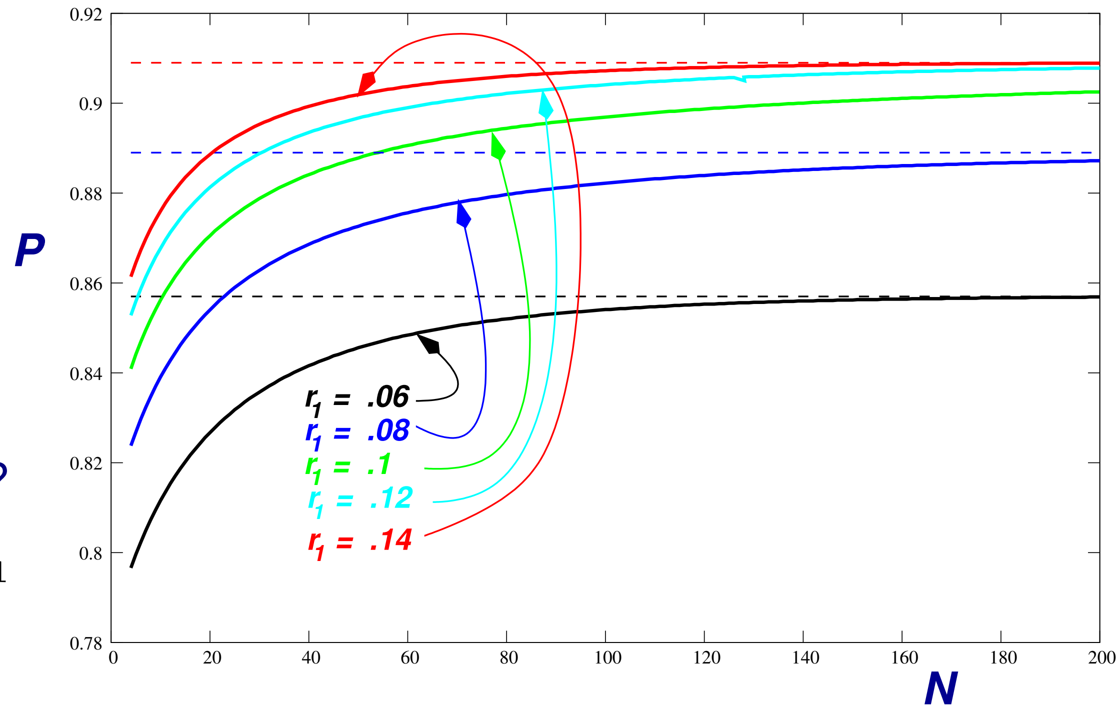
$\tau = 1.$
$p_1 = .01$
$r_2 = .1$
$p_2 = .01$



Finite-Buffer Lines



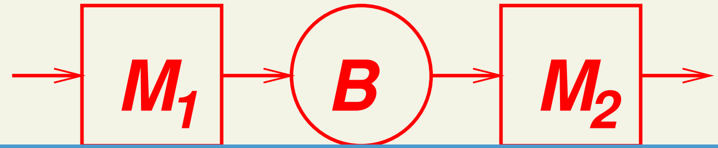
Production rate vs. Buffer Size



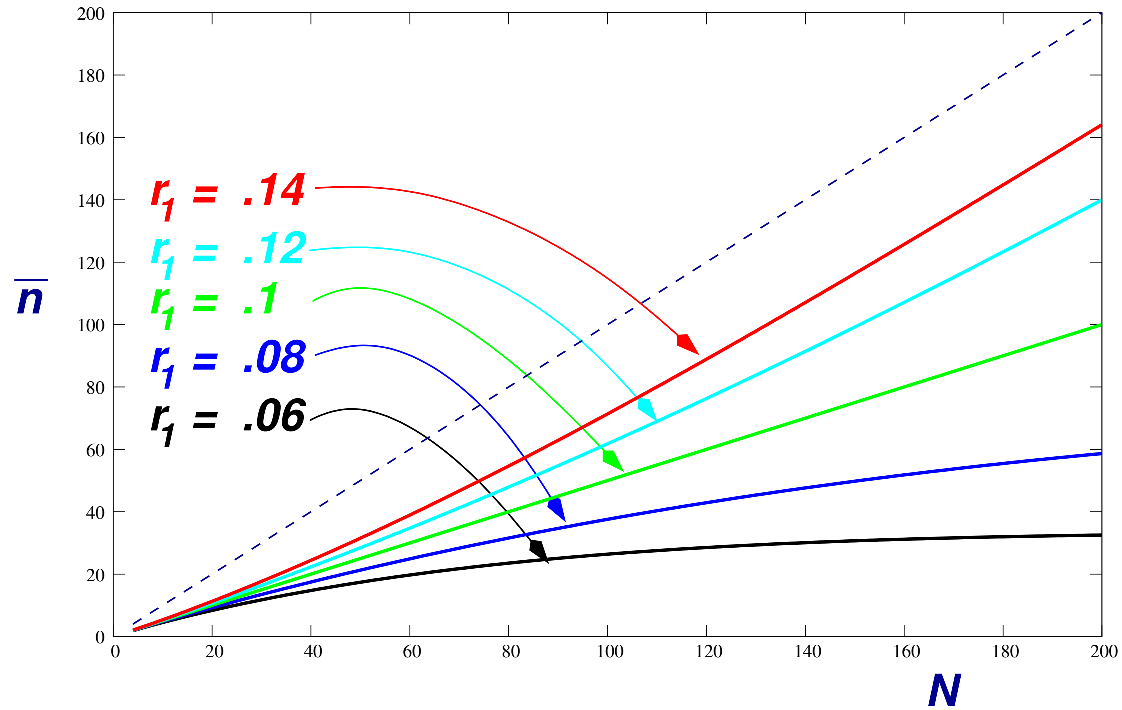
Discussion:

- Why are the curves increasing?
- Why do they reach an asymptote?
- What is P when $N = 0$?
- What is the limit of P as $N \to \infty$?
- Why are the curves with smaller r_1 lower?

Finite-Buffer Lines



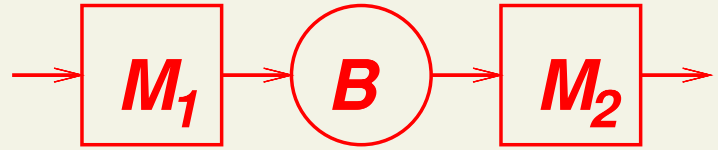
Average Inventory vs. Buffer Size

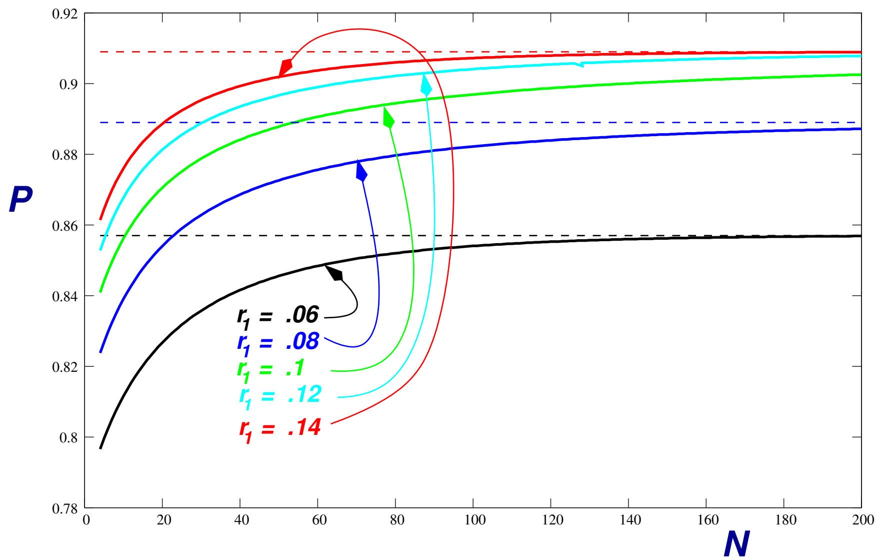


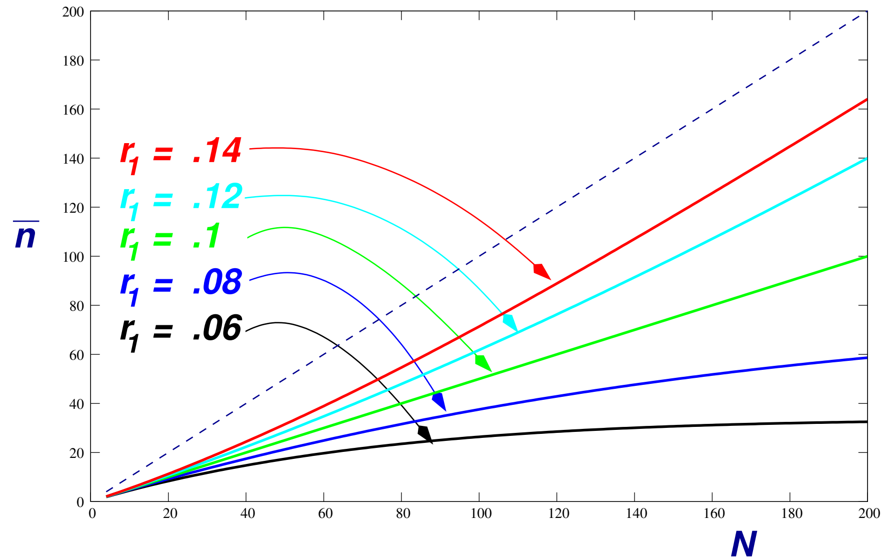
Discussion:

- Why are the curves increasing?
- Why *different* asymptotes?
- What is $\bar{n}$ when $N = 0$?
- What is the limit of $\bar{n}$ as $N \to \infty$?
- Why are the curves with smaller r_1 lower?

Finite-Buffer Lines

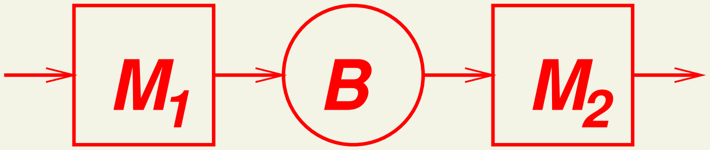






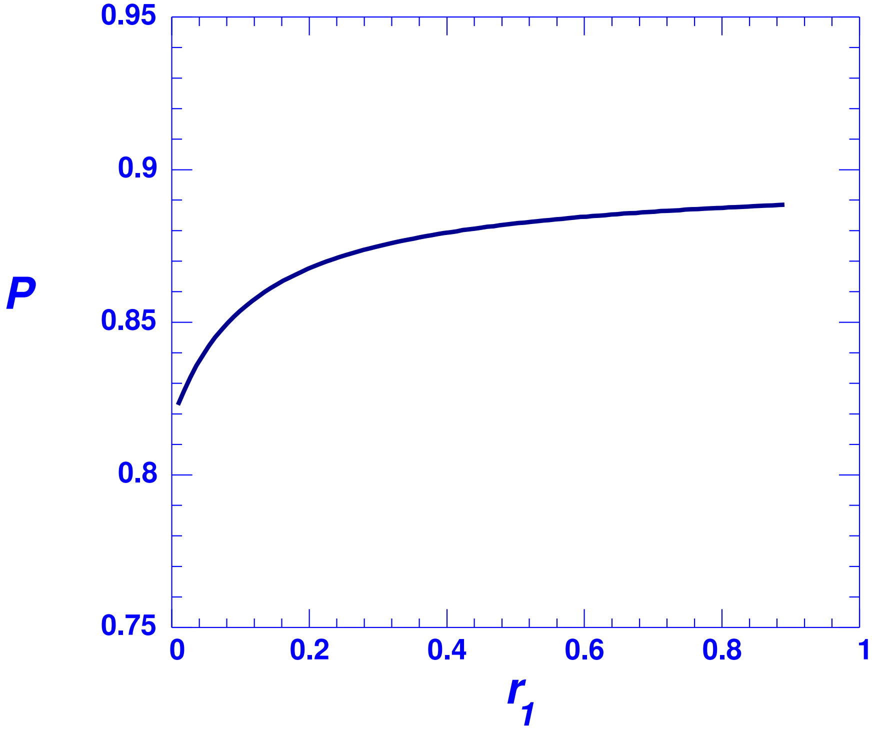
- *What can you say about the optimal buffer size?*
- *How should it be related to r_i, p_i?*

Finite-Buffer Lines

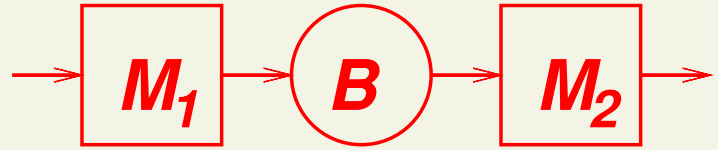


Should we prefer short, frequent, disruptions or long, infrequent, disruptions?

- $r_2 = 0.8$, $p_2 = 0.09$, $N = 10$
- r_1 and p_1 vary together and $\frac{r_1}{r_1+p_1} = .9$
- *Answer:* evidently, short, frequent failures.
- *Why?*



Finite-Buffer Lines



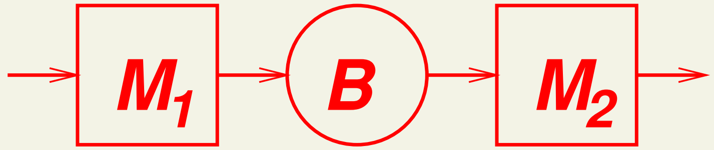
Questions:

- If we want to increase production rate, which machine should we improve?
- What would happen to production rate if we improved any other machine?

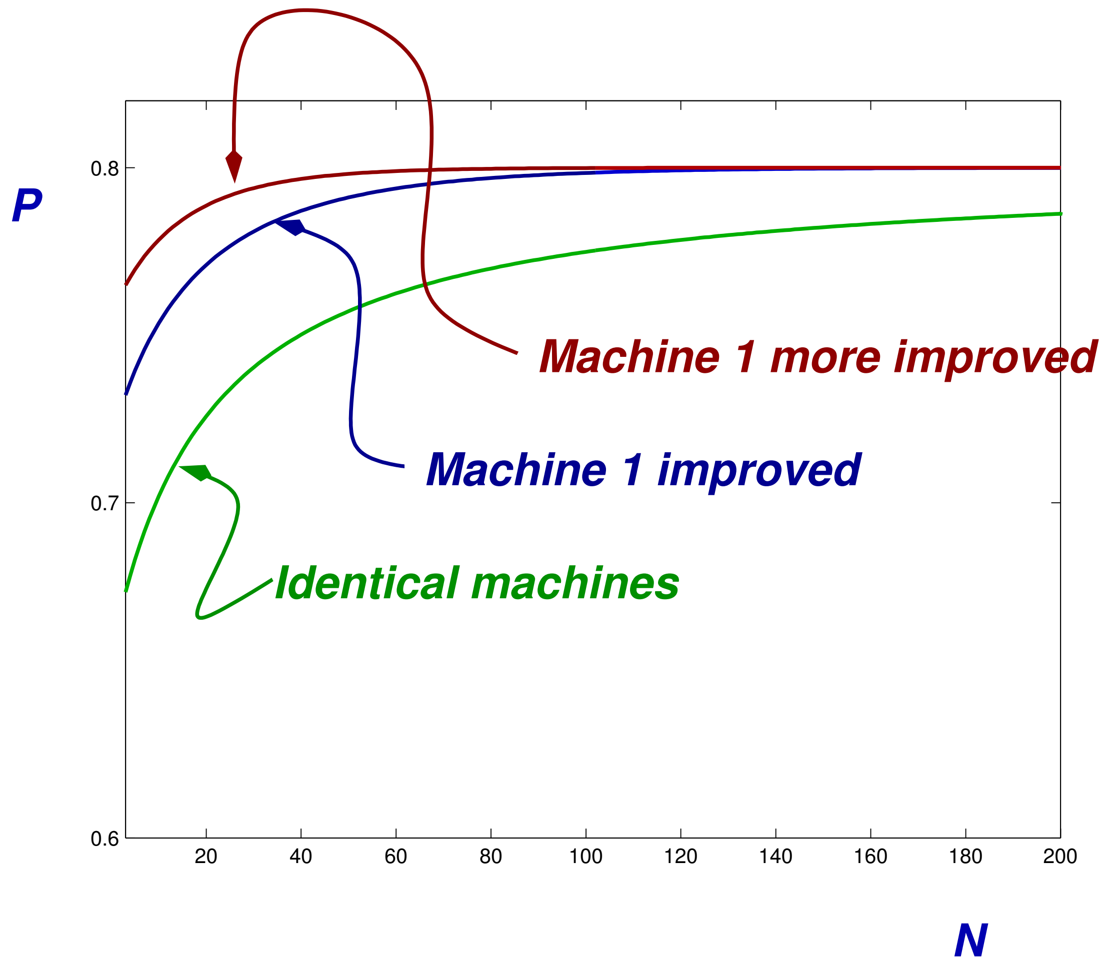
Discussion:

- Is randomness the main issue here, or can it be summarized by an average?

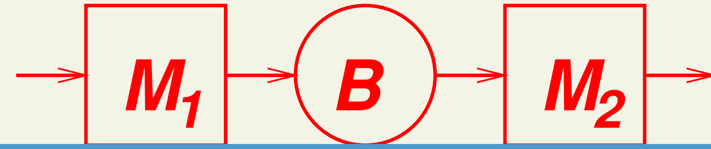
Finite-Buffer Lines



Improvements to *non-bottleneck* machine.

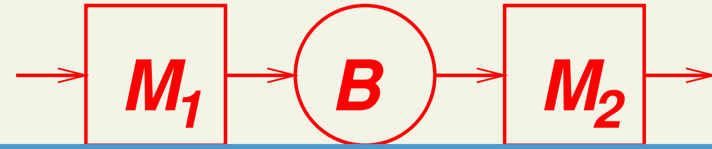


Finite-Buffer Lines



Exponential — discrete material, continuous time

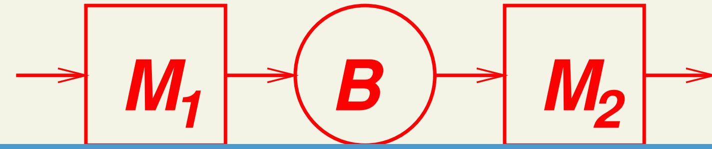
- $\mu_i \delta t$ = the probability that M_i completes an operation in $(t, t + \delta t)$;
- $p_i \delta t$ = the probability that M_i fails during an operation in $(t, t + \delta t)$;
- $r_i \delta t$ = the probability that M_i is repaired, while it is down, in $(t, t + \delta t)$;



Continuous — continuous material, continuous time

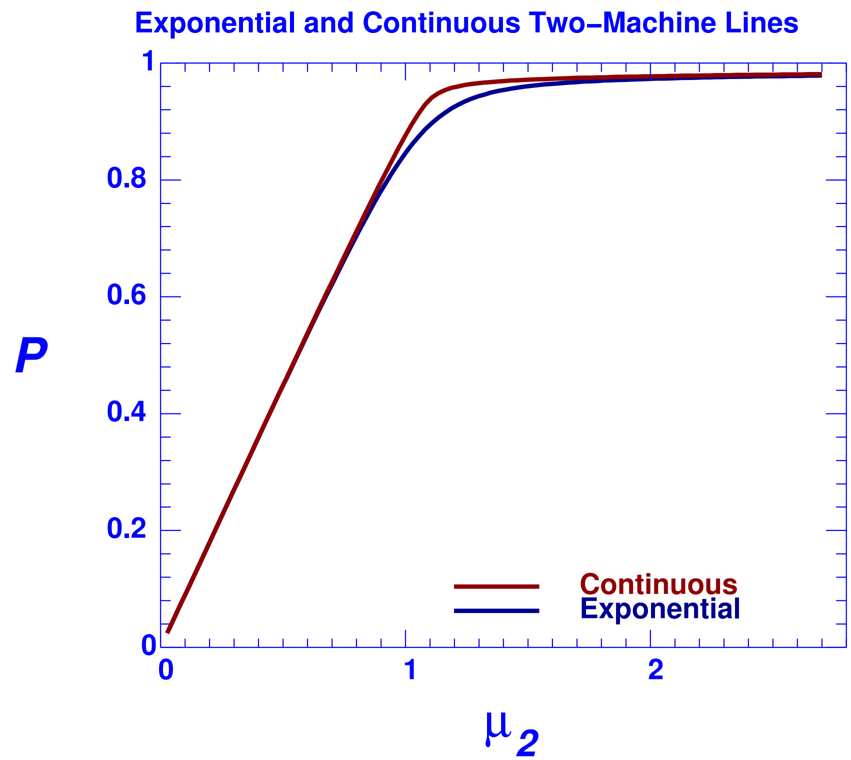
- $\mu_i \delta t =$ the amount of material that M_i processes, while it is up, in $(t, t + \delta t)$;
- $p_i \delta t =$ the probability that M_i fails, while it is up, in $(t, t + \delta t)$;
- $r_i \delta t =$ the probability that M_i is repaired, while it is down, in $(t, t + \delta t)$;

Finite-Buffer Lines

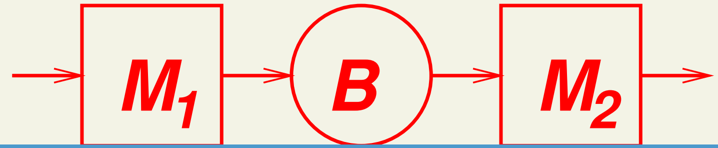


Other models

- $r_1 = 0.09$, $p_1 = 0.01$, $\mu_1 = 1.1$
- $r_2 = 0.08$, $p_1 = 0.009$
- $N = 20$
- *Explain the shapes of the graphs.*

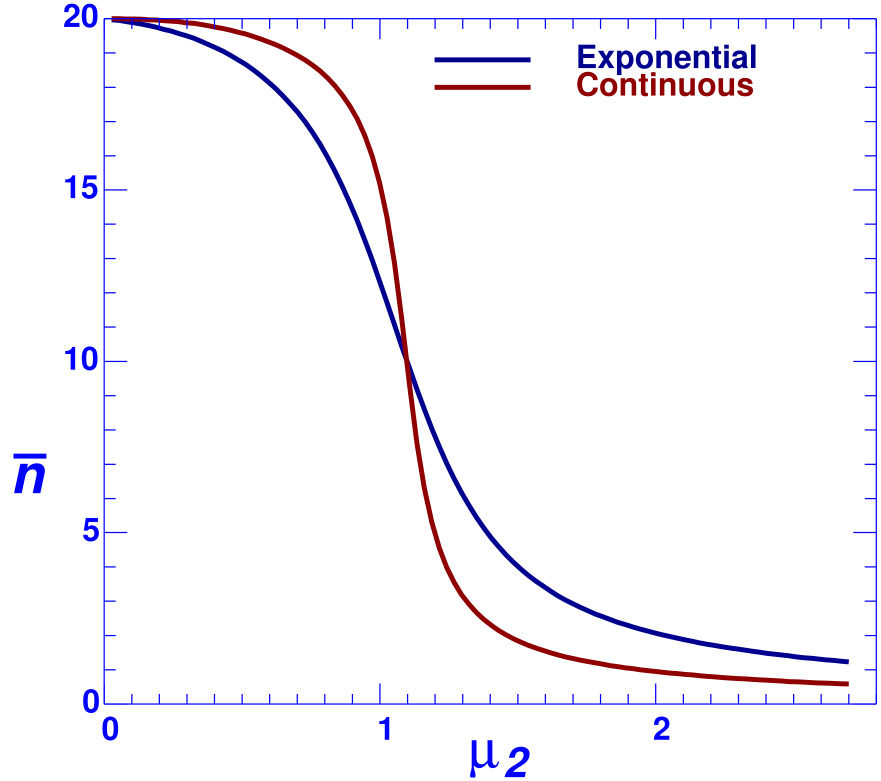


Finite-Buffer Lines

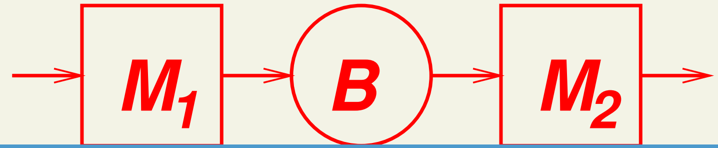


Other models

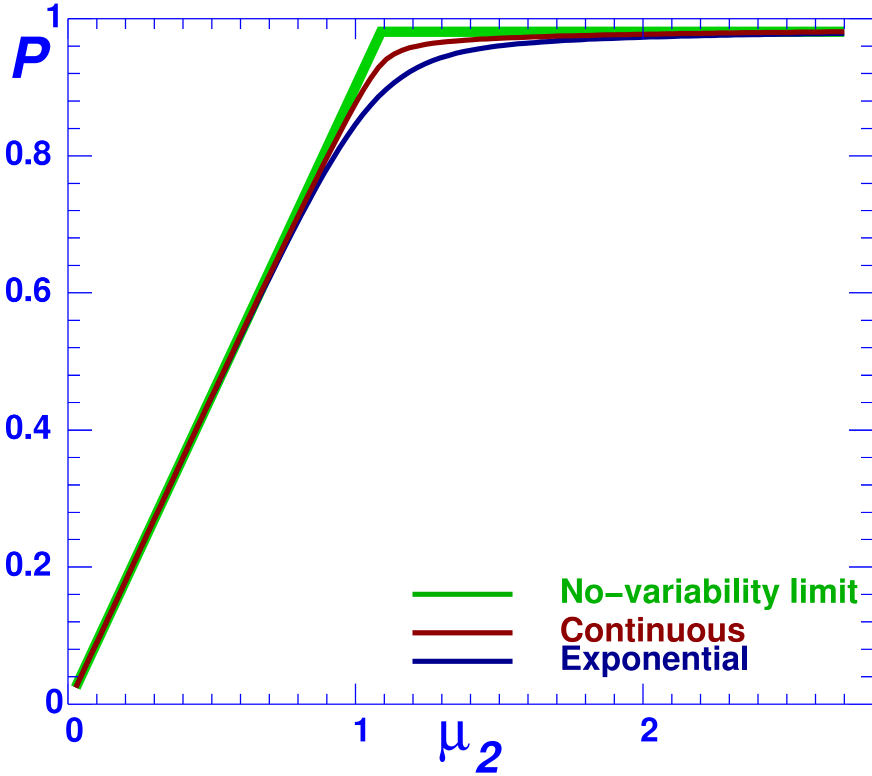
- *Explain the shapes of the graphs.*

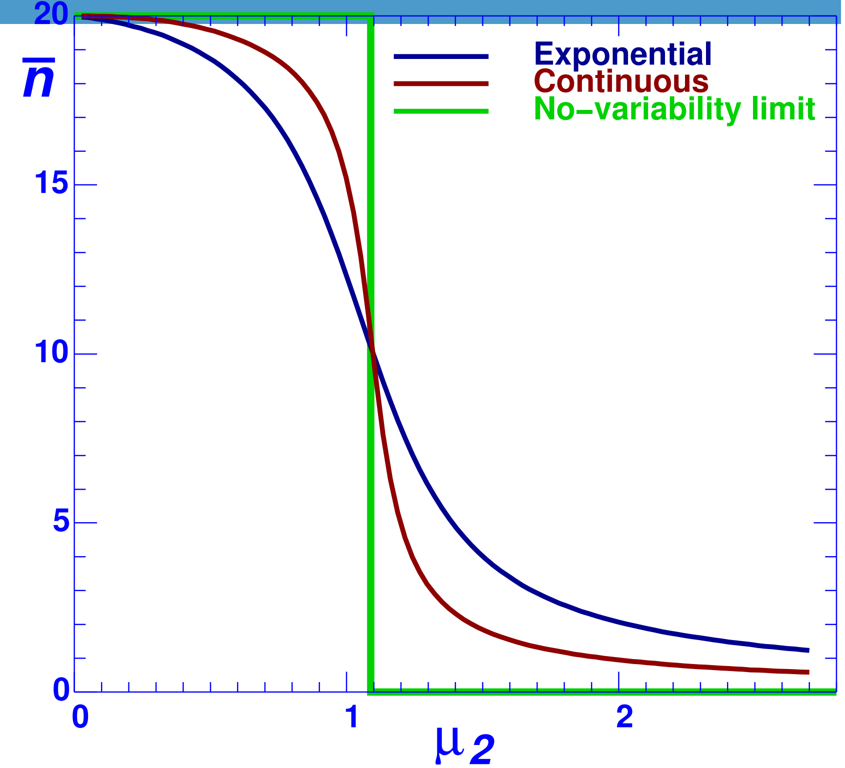


Finite-Buffer Lines



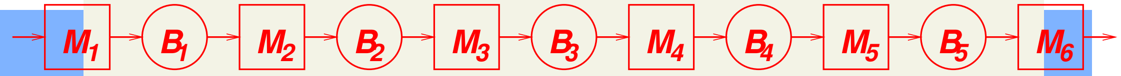
Other models



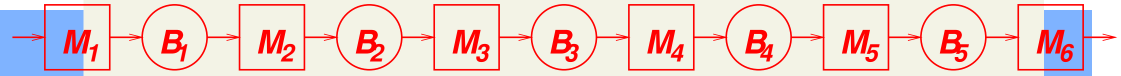


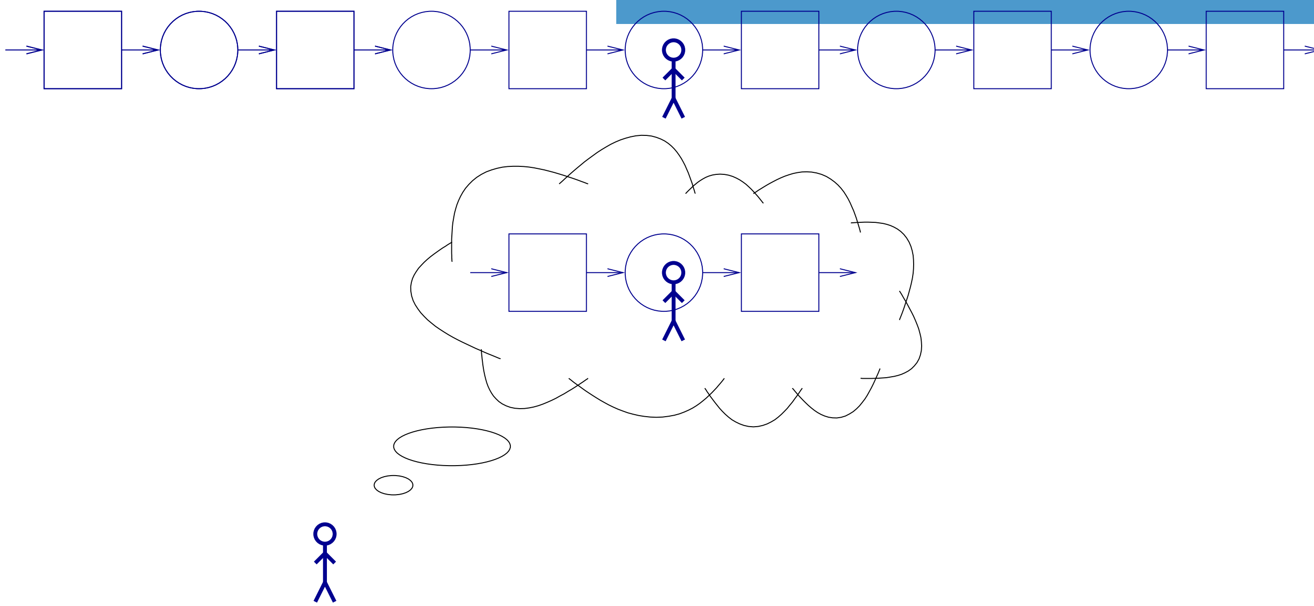
No-variability limit: a continuous model where both machines are reliable, and processing rate μ_i' of machine i in the no-variability is the same as the isolated production rate of machine i in the other cases. That is, $\mu_i' = \mu_i r_i / (r_i + p_i)$.

Finite-Buffer Lines



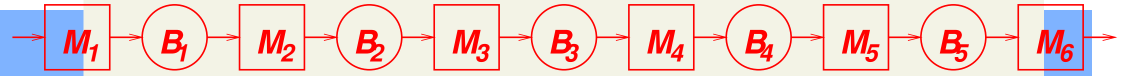
- Difficulty:
  - ⋆ No simple formula for calculating production rate or inventory levels.
  - ⋆ State space is too large for exact numerical solution.
    - ∗ If all buffer sizes are N and the length of the line is k, the number of states is $S = 2^k(N+1)^{k-1}$.
    - ∗ if $N = 10$ and $k = 20$, $S = 6.41 \times 10^{25}$.
  - ⋆ *Decomposition* seems to work successfully.



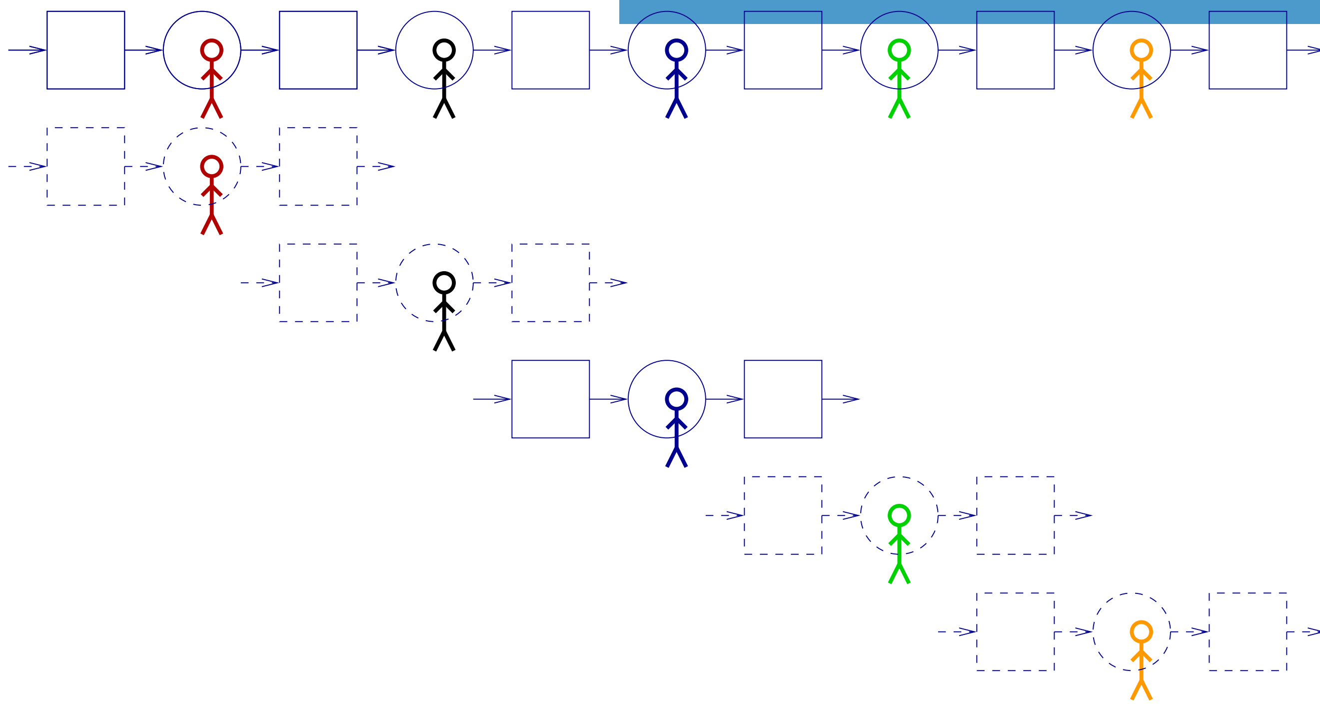


- Conceptually: put an observer in a buffer, and tell him that he is in the buffer of a two-machine line.
- Question: *What would the observer see, and how can he be convinced he is in a two-machine line?*

Finite-Buffer Lines

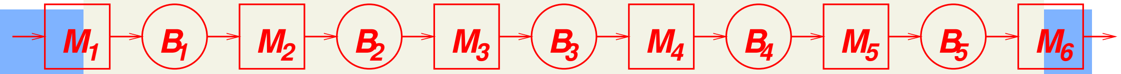


Decomposition



- Decomposition breaks up systems and then reunites them.
- *Construct all the two-machine lines.*

Finite-Buffer Lines

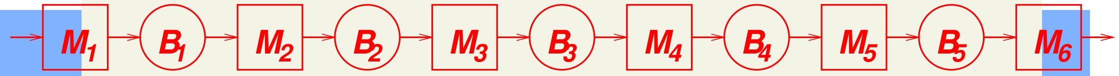


Decomposition

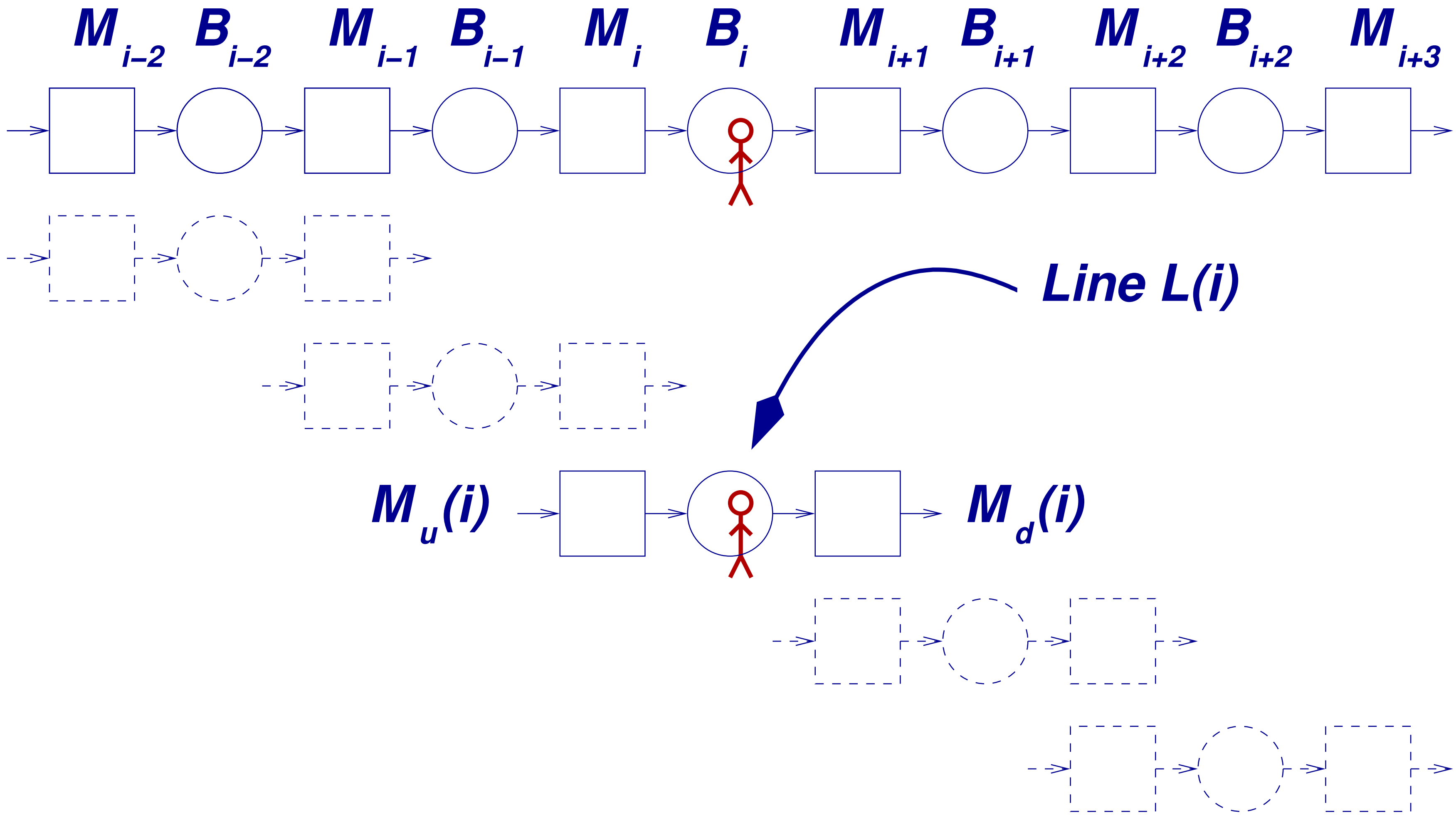
- Consider an observer in Buffer B_i.
  - ⋆ Imagine the material flow process that the observer sees *entering* and the material flow process that the observer sees *leaving* the buffer.
- We construct a two-machine line $L(i)$
  - ⋆ ie, we find machines $M_u(i)$ and $M_d(i)$ with parameters $r_u(i)$, $p_u(i)$, $r_d(i)$, $p_d(i)$, and $N(i) = N_i$)

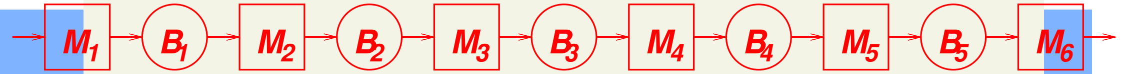
  such that an observer in its buffer will see almost the same processes.
- The parameters are chosen as functions of the behaviors of the *other* two-machine lines.

Finite-Buffer Lines



Decomposition





There are $4(k-1)$ unknowns ($r_u(i+1)$, $p_u(i+1)$, $r_d(i)$, $p_d(i)$, $i = 1, ..., k-1$). Therefore, we need

- $4(k-1)$ equations, and
- an algorithm for solving those equations.

Finite-Buffer Lines

Equations

- *Conservation of flow,* equating all production rates.
- *Flow rate/idle time,* relating production rate to probabilities of starvation and blockage.
- *Resumption of flow,* relating $r_u(i)$ to upstream events and $r_d(i)$ to downstream events.
- *Boundary conditions,* for parameters of $M_u(1)$ and $M_d(k-1)$.

Finite-Buffer Lines

Equations

Conservation of Flow

$$P(i) = P(1), i = 2, \ldots, k-1.$$

Note that

$$P(i) = P(r_u(i), p_u(i), r_d(i), p_d(i), N(i))$$

This is a set of $k - 2$ equations.

Finite-Buffer Lines

Equations

Flow Rate-Idle Time

$$P_i = e_i \text{ prob } [n_{i-1} > 0 \text{ and } n_i < N_i]$$

where

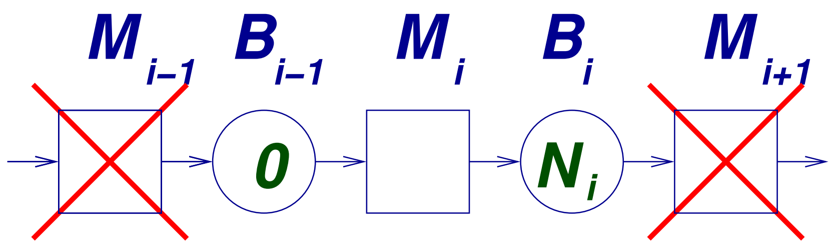
$$e_i = \frac{r_i}{r_i + p_i}$$

Problem: this expression involves a joint probability of two buffers taking certain values. But we only know how to calculate the probability of one buffer taking on a value at a time.

Observation:

$$\text{prob}\ (n_{i-1} = 0 \text{ and } n_i = N_i) \approx 0.$$

Reason:



The only way to have $n_{i-1} = 0$ and $n_i = N_i$ is if

- M_{i-1} is down or starved for a long time
- *and* M_i is up
- *and* M_{i+1} is down or blocked for a long time
- *and* to have exactly N_i parts in the two buffers.

Then

$$\text{prob } [n_{i-1} > 0 \text{ and } n_i < N_i]$$

$$\approx 1 - \{ \text{prob } (n_{i-1} = 0) + \text{ prob } (n_i = N_i)\}$$

Therefore

$$P_i \approx e_i \left[1 - \text{ prob } (n_{i-1} = 0) - \text{ prob } (n_i = N_i)\right]$$

or

$$P(i) \approx e_i \left[1 - p_s(i-1) - p_b(i)\right]$$

Finite-Buffer Lines

Equations

Flow Rate-Idle Time

Note that

$$P(i) = e_u(i)\,[1 - p_b(i)]$$
$$P(i-1) = e_d(i-1)\,[1 - p_s(i-1)]$$

so the previous expression can be transformed to

$$\boxed{\frac{p_d(i-1)}{r_d(i-1)} + \frac{p_u(i)}{r_u(i)} = \frac{1}{P(i)} + \frac{1}{e_i} - 2, i = 2, \ldots, k-1}$$

This is a set of $k - 2$ equations.

This is a long derivation, so we will jump to the equations:

$$X(i) = \frac{p_s(i-1) r_u(i)}{p_u(i) E(i)}$$

and

$$r_u(i) = r_u(i-1) X(i) + r_i (1 - X(i)), i = 2, \ldots, k-1$$

This is a set of $k - 2$ equations.

Finite-Buffer Lines

Equations

Resumption of Flow

And,

$$r_d(i-1) = r_d(i)Y(i) + r_i(1 - Y(i)), i = 2, \ldots, k-1$$

where

$$Y(i) = \frac{p_b(i) r_d(i-1)}{p_d(i-1) E(i-1)}.$$

This is a set of $k-2$ equations. We now have $4(k-2) = 4k-8$ equations.

$M_d(1)$ is the same as M_1 and $M_d(k-1)$ is the same as M_k. Therefore

$$\begin{aligned} r_u(1) &= r_1 \\ p_u(1) &= p_1 \\ r_d(k-1) &= r_k \\ p_d(k-1) &= p_k \end{aligned}$$

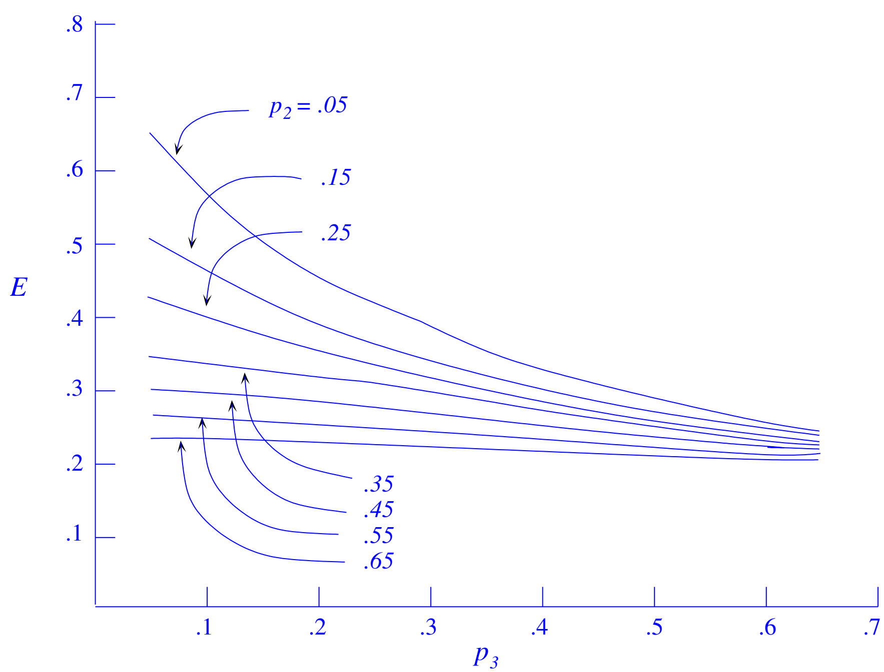
This is a set of 4 equations. We now have $4(k-1)$ equations in $4(k-1)$ unknowns $r_u(i)$, $p_u(i)$, $r_d(i)$, $p_d(i)$, $i = 1, ..., k-1$.

- All the quantities in these equations are
  - ⋆ specified parameters, or
  - ⋆ unknowns, or
  - ⋆ functions of parameters or unknowns derived from the two-machine line analysis.
- This is a set of $4(k-1)$ equations in the same number of unknowns.

DDX algorithm : due to Dallery, David, and Xie (1988).

1. Guess the downstream parameters of $L(1)$ $(r_d(1), p_d(1))$. Set $i = 2$.
2. Use the equations to obtain the upstream parameters of $L(i)$ $(r_u(i), p_u(i))$. Increment i.
3. Continue in this way until $L(k-1)$. Set $i = k-2$.
4. Use the equations to obtain the downstream parameters of $L(i)$. Decrement i.
5. Continue in this way until $L(1)$.
6. Go to Step 2 or terminate.

Three-machine line – production rate.

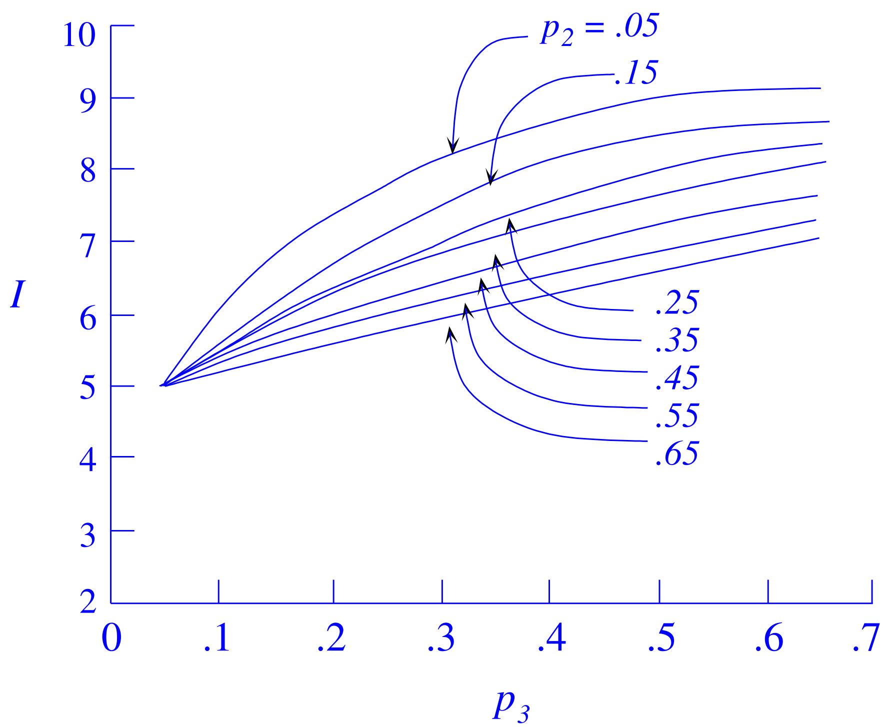


$$r_1 = r_2 = r_3 = .2$$

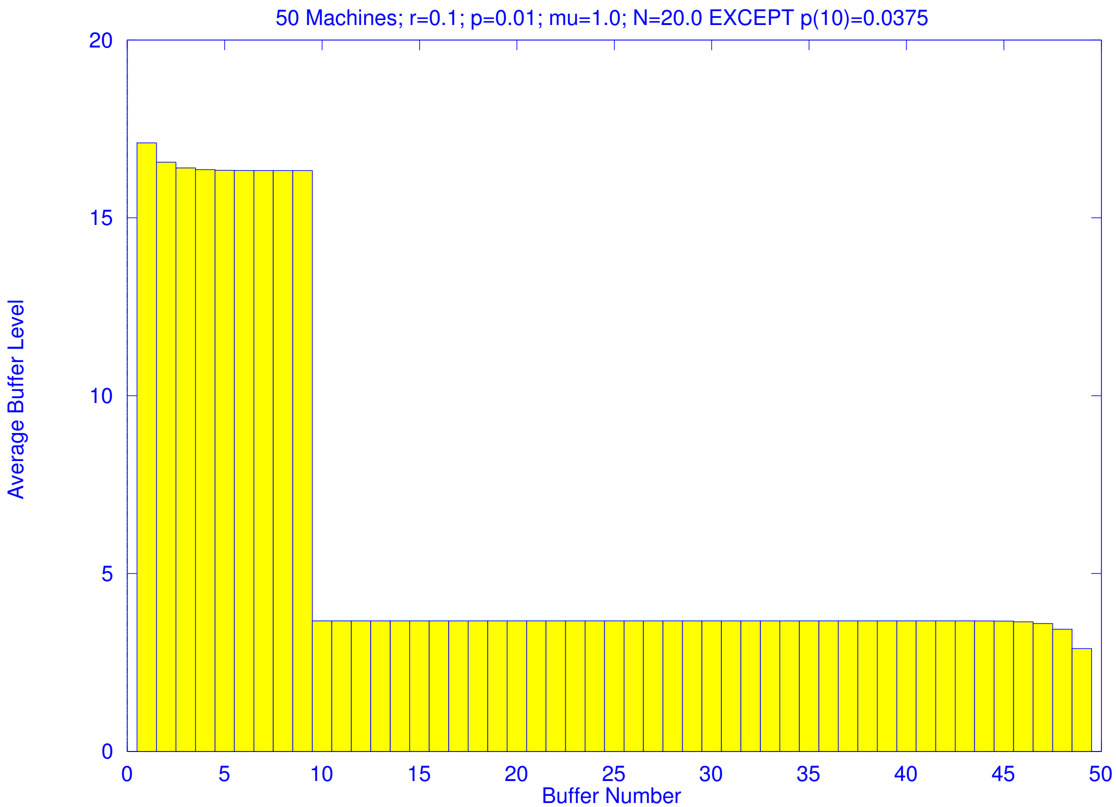
$$p_1 = .05$$

$$N_1 = N_2 = 5$$

Three-machine line – total average inventory



$$r_1 = r_2 = r_3 = .2$$
$$p_1 = .05$$
$$N_1 = N_2 = 5$$



Effect of a bottleneck. Identical machines and buffers, except for M_{10}.

Finite-Buffer Lines

Discussion

- Explain the shape.

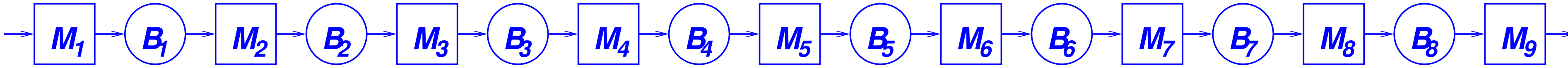
- Is randomness the main issue here, or can it be summarized by an average?
  - ⋆ *What features of the graph are determined by the variability?*
  - ⋆ *What features of the graph are determined by the bottleneck?*

Finite-Buffer Lines

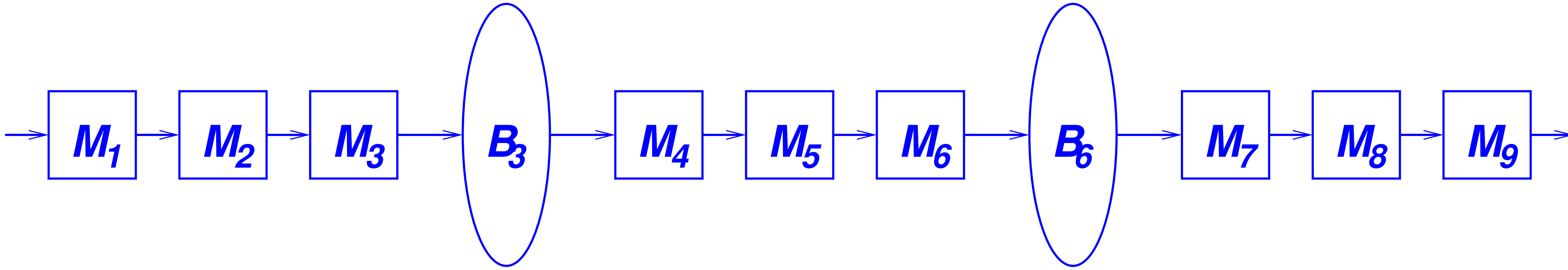
Discussion

Which has a higher production rate?

- 9-Machine line with two buffering options:
  - ⋆ 8 buffers equally sized; and

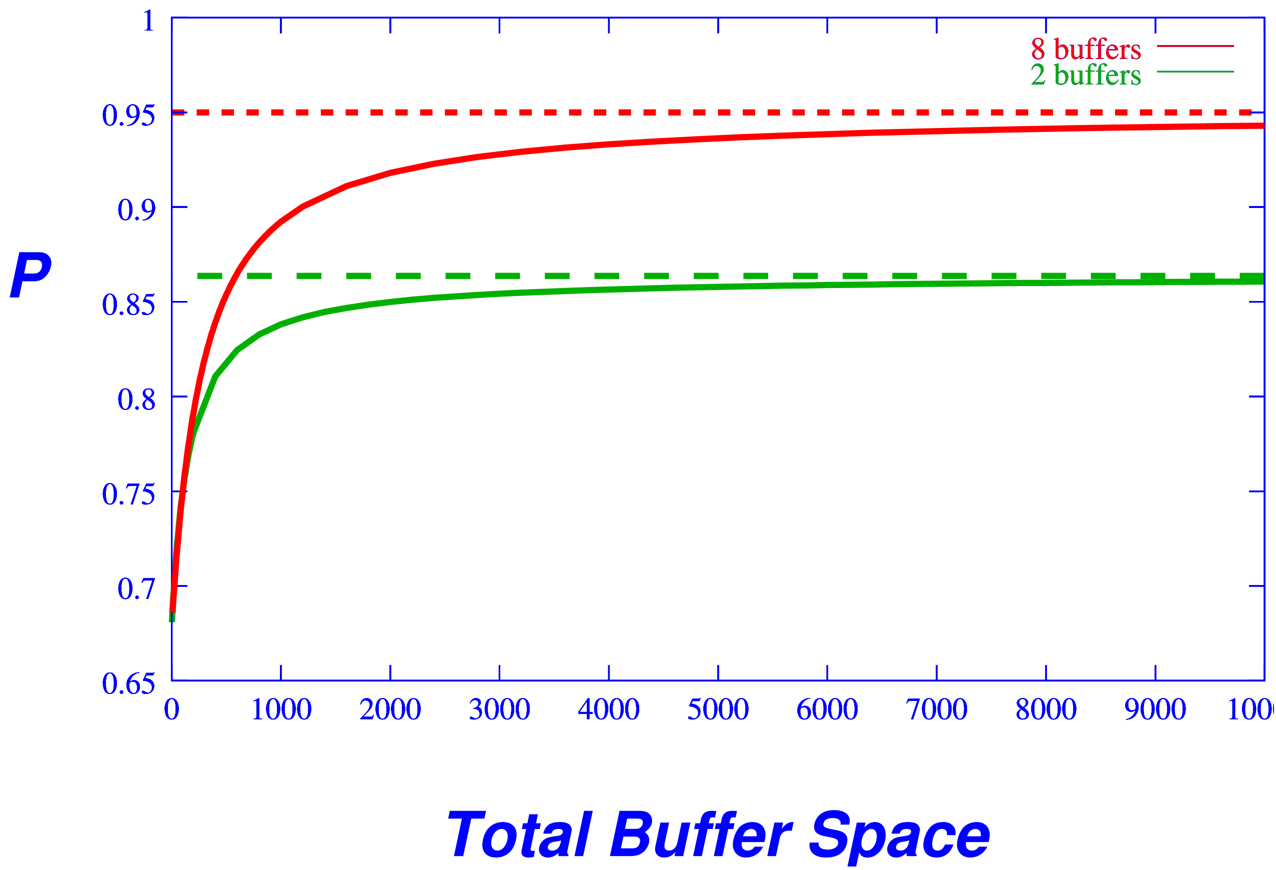


  - ⋆ 2 buffers equally sized.



Finite-Buffer Lines

Discussion



- Continuous model; all machines have $r = .019$, $p = .001$, $\mu = 1$.
- What are the asymptotes?
- Is 8 buffers *always* faster?

Finite-Buffer Lines

Discussion

Optimal buffer space distribution.

- Design the buffers for a 20-machine production line.
- The machines have been selected, and the only decision remaining is the amount of space to allocate for in-process inventory.
- *The goal is to determine the smallest amount of in-process inventory space so that the line meets a production rate target.*

Finite-Buffer Lines

Discussion

Optimal buffer space distribution.

- The common operation time is one operation per minute.
- The target production rate is .88 parts per minute.

- *Case 1* MTTF= 200 minutes and MTTR = 10.5 minutes for all machines ($\boldsymbol{P = .95}$ parts per minute).

- *Case 1* MTTF= 200 minutes and MTTR = 10.5 minutes for all machines ($P = .95$ parts per minute).
- *Case 2* Like Case 1 except Machine 5. For Machine 5, MTTF = 100 and MTTR = 10.5 minutes ($P = .905$ parts per minute).

- *Case 1* MTTF= 200 minutes and MTTR = 10.5 minutes for all machines ($P = .95$ parts per minute).
- *Case 2* Like Case 1 except Machine 5. For Machine 5, MTTF = 100 and MTTR = 10.5 minutes ($P = .905$ parts per minute).
- *Case 3* Like Case 1 except Machine 5. For Machine 5, MTTF = 200 and MTTR = 21 minutes ($P = .905$ parts per minute).

Are buffers really needed?

Line	Production rate with no buffers, parts per minute
Case 1	.487
Case 2	.475
Case 3	.475

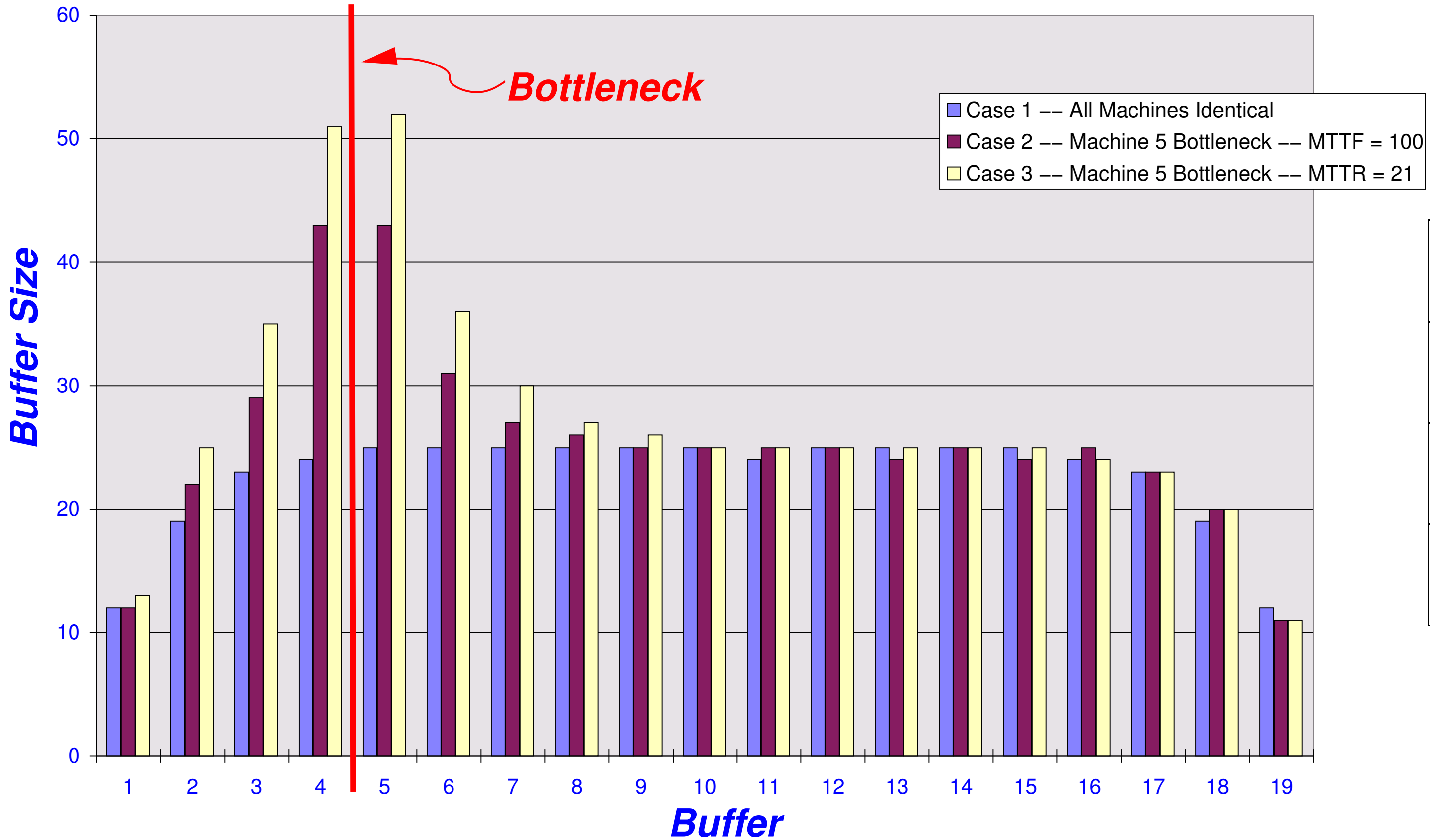
Yes. *How were these numbers calculated?*

Finite-Buffer Lines

Discussion

Optimal buffer space distribution.

Solution



Line	Space
Case 1	430
Case 2	485
Case 3	523

Finite-Buffer Lines

Discussion

Optimal buffer space distribution.

- Observation from studying buffer space allocation problems:
  - ⋆ *Buffer space is needed most where buffer level variability is greatest!*

HP Printer Case

Background

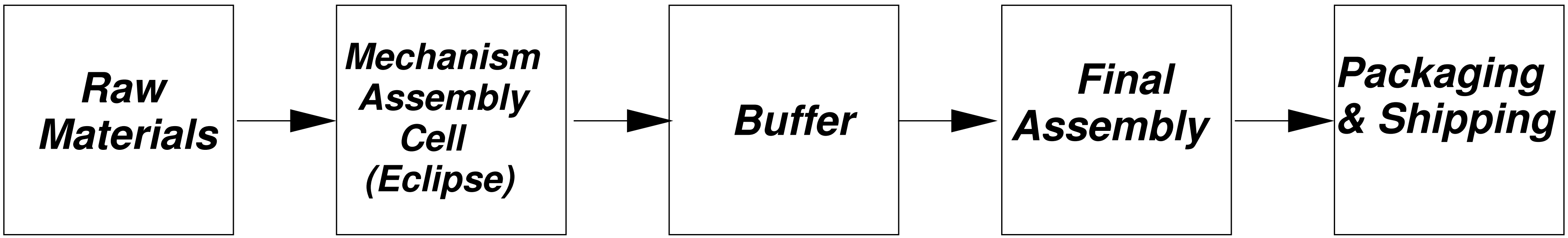
- In 1993, the ink-jet printer market was taking off explosively, and manufacturers were competing intensively for market share.
- Manufacturers could sell all they could produce. Demand was much greater than production capacity.
- Hewlett Packard was designing and producing its printers in Vancouver, Washington (near Portland, Oregon).

HP Printer Case

- Maintain quality.
- Meet increased demand *and* increase market share.
  - ⋆ *Target: 300,000 printers/month.*
- Meet profit and revenue targets.
- Keep employment stable.
  - ⋆ *Capacity with existing manual assembly: 200,000 printers/month.*

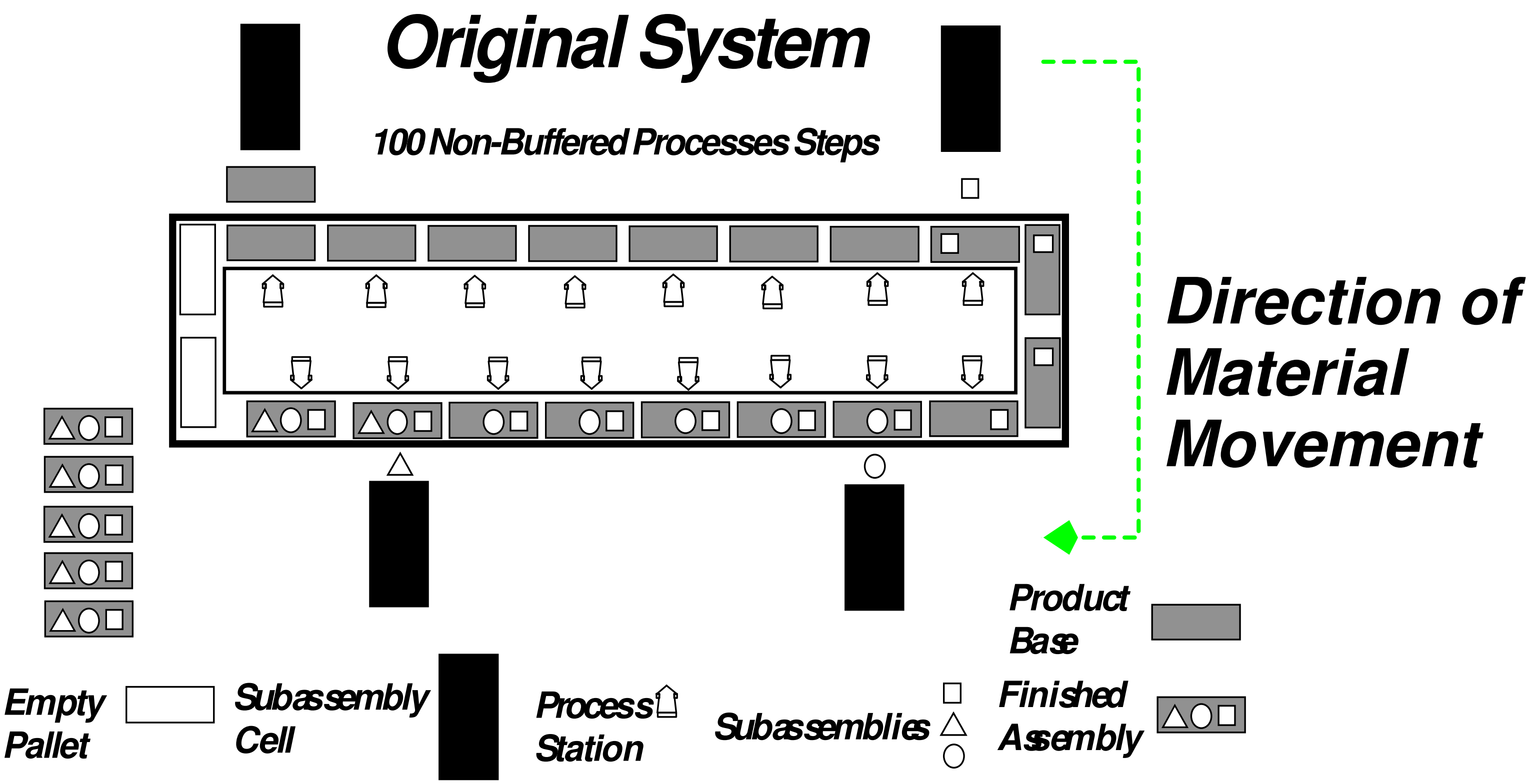
HP Printer Case

HP invested $25,000,000 in “Eclipse,” a new system for automated assembly of the print engine.



Two Eclipses were installed.

HP Printer Case



Design philosophy: minimal — essentially zero — buffer space.

To estimate the production capacity, recall the formula for machine efficiency,

$$e = \frac{r}{r+p} = \frac{1}{1+\frac{p}{r}}$$

and Buzacott's zero-buffer formula,

$$P = \frac{1}{\tau} \frac{1}{1 + \sum_{i=1}^{k} \frac{p_i}{r_i}}$$

HP Printer Case

Printer Production

Estimated capacity

- Machine efficiencies e are all about .99. Therefore $p_i/r_i \approx .01$.
- k = 50 machines. (More than one process step per machine.)
- $1/\tau = 400$ units per hour, 685 available hours per month
- yield = .995

Therefore the total production rate is

$$P = (.995)(2)(400)(685)\frac{1}{1 + (50)(.01)}$$

$$\approx 369,900 \text{ units per month}$$

which is greater than the target, so the target appears to be feasible. (Note: the production rate is extremely sensitive to e.)

HP Printer Case

The Problem

Data was collected when the first two machines were installed.

- Efficiency was less than .99.
- Operation times were variable and often greater than 3600/400=9 seconds.

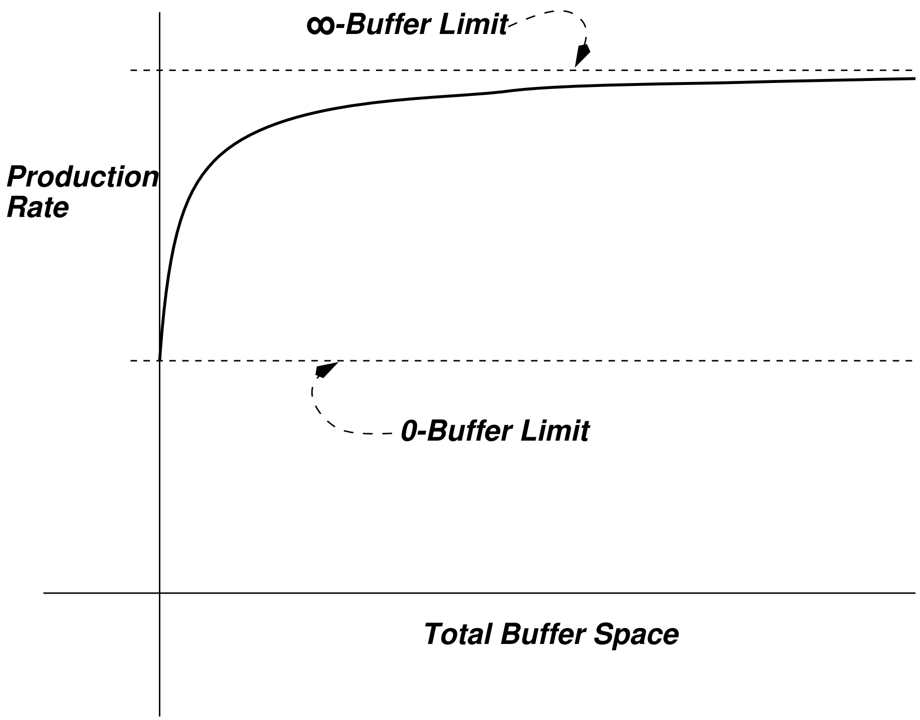
Therefore actual production rate would be about 125,000 units/month, much less than the target, if the design of the system were not changed.

HP Printer Case

- HP tried to analyze the system by simulation. They consulted a vendor, but the project appeared to be too large and complex to produce useful results in time to affect the system design.
  - ⋆ *This was because they tried to include too much detail.*
- Infeasible changes: adding labor, redesigning machines.

HP Printer Case

- Feasible change: adding buffer space within Eclipse.

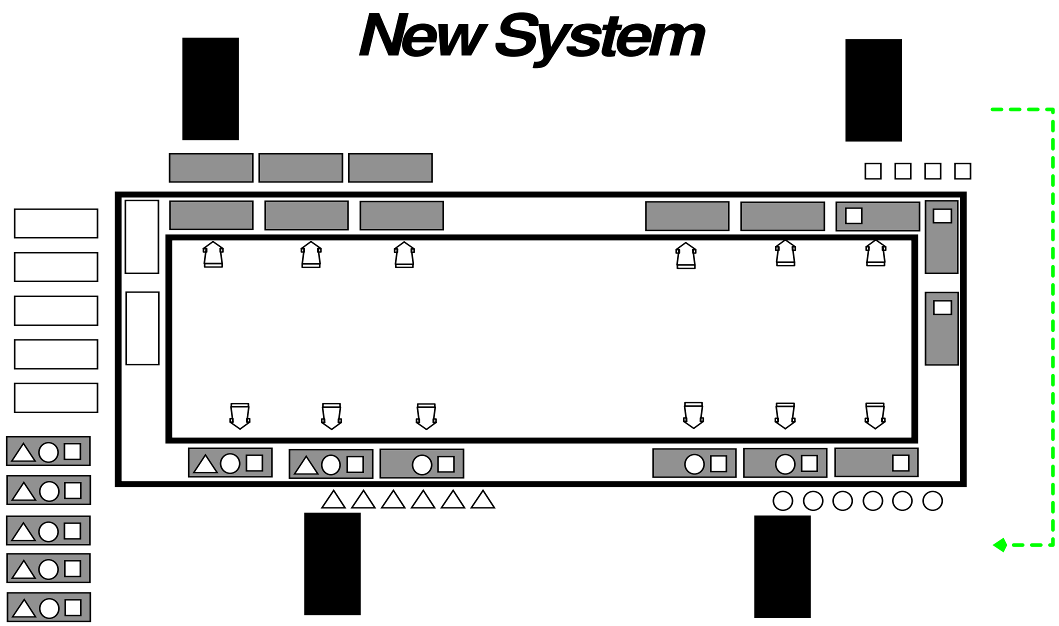


- Design tool: decomposition.

HP Printer Case

- The existing decomposition method did not have certain features of the Eclipse system:
  - ⋆ Machines with different operation times;
  - ⋆ Closed loop pallet flow.
- This required creative approximation.
- Both of these features were added in later research.

HP Printer Case



- Empty pallet buffer.
- WIP space between subassembly lines and main line.
- WIP space on main line.
- Buffer sizes were large enough to hold about 30 minutes worth of material. This is a small multiple of the mean time to repair (MTTR) of the machines.

HP Printer Case

- Increased factory capacity — to over 250,000 units/month.
- Capital cost of changes was about $1,400,000.
- Incremental revenues of about $280,000,000.
- Labor productivity increased by about 50%.
- Improved factory design method.
- New research results which have been incorporated in courses.
- MIT spin-off (Analytics Operations Engineering, Inc.).

HP Printer Case

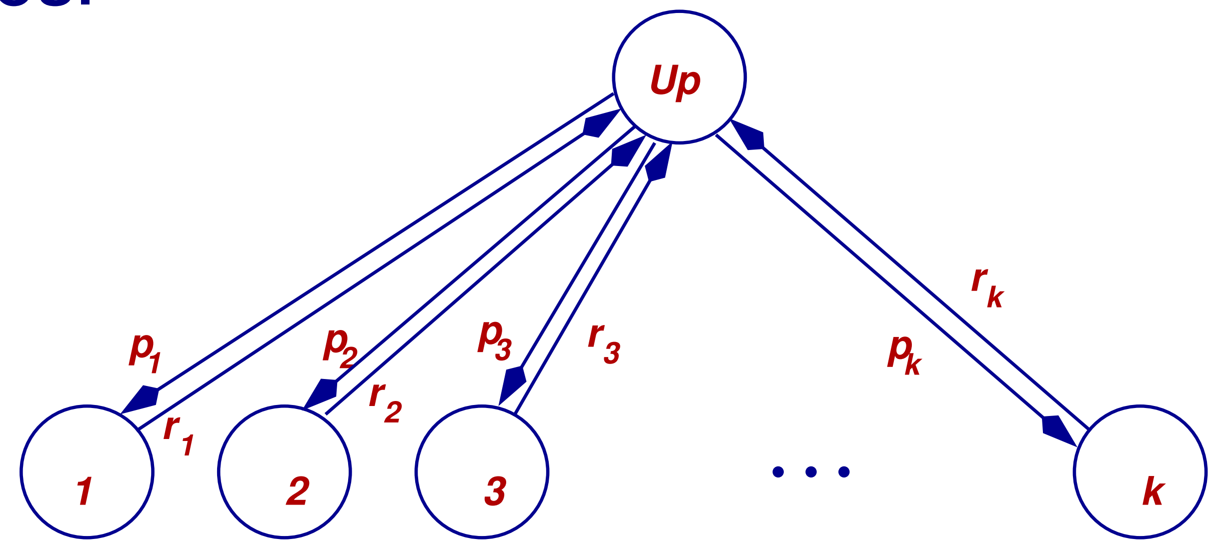
- Early intervention.
- Rapid response by MIT researchers because much related work already done.
- HP managers' flexibility.
- The new analysis tool was fast, easy to use, and was at the right level of detail.

Tolio Decomposition

- *Goal:* To improve accuracy of decomposition when machines have very different repair times.
- *Goal:* To allow machines to have multiple failure modes.
- *Additional benefit:* To extend decomposition to systems with closed loops.

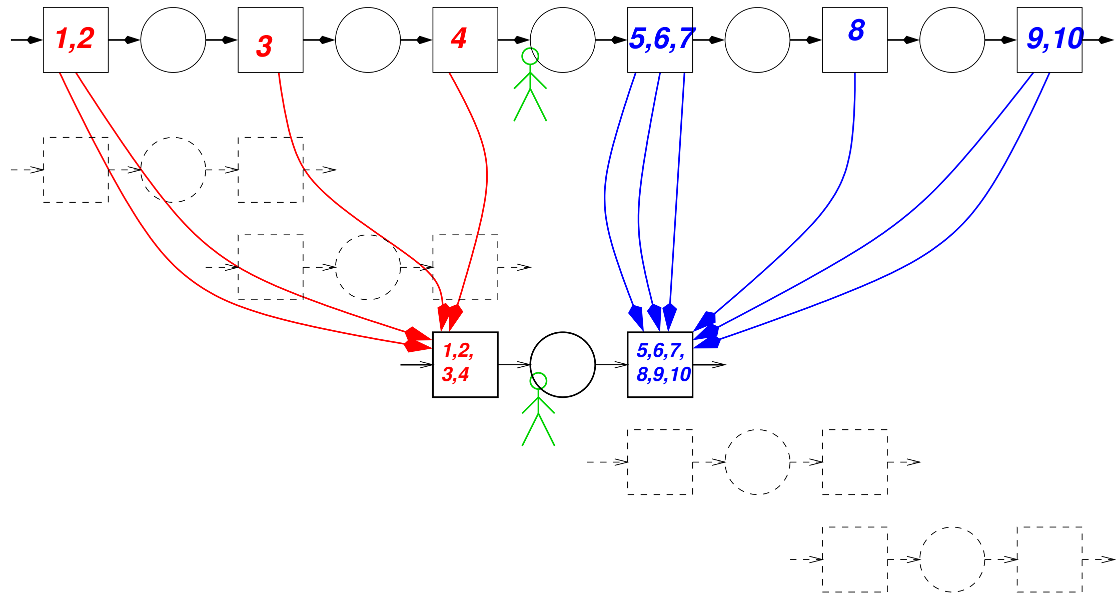
Tolio Decomposition

First, extend single machine model to having multiple down states.



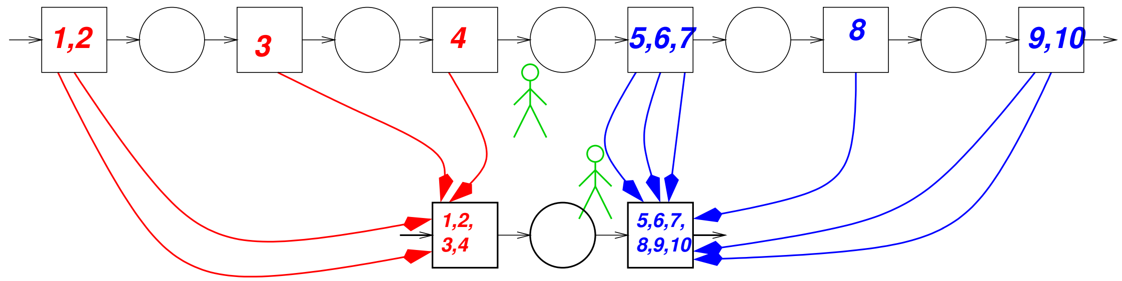
Then, extend two-machine line so that both machines have multiple down states.

Tolio Decomposition



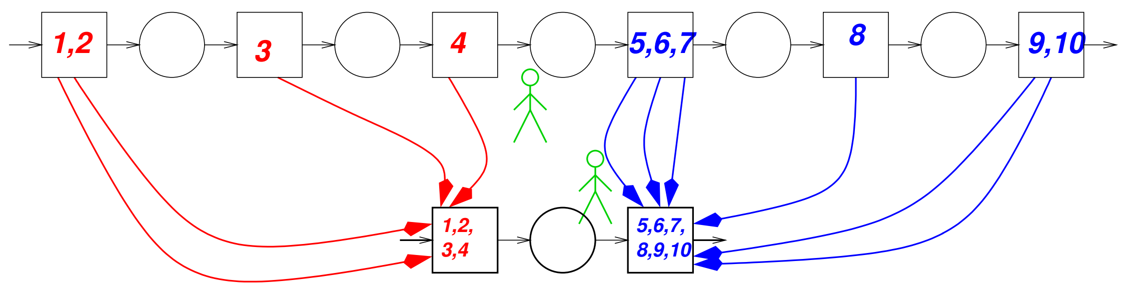
- There is an observer in each buffer who is told that he is actually in the buffer of a two-machine line.

Tolio Decomposition



- Each machine in the original line *may* and in the two-machine lines *must* have multiple failure modes.
- For each failure mode downstream of a given buffer, there is a corresponding mode in the downstream machine of its two-machine line.
- Similarly for upstream modes.

Tolio Decomposition



- The downstream failure modes appear to the observer after propagation through *blockage* .
- The upstream failure modes appear to the observer after propagation through *starvation* .
- The two-machine lines are more complex that in earlier decompositions but the decomposition equations are simpler.

Tolio Decomposition

- A set of decomposition equations are formulated.
- They are solved by a Dallery-David-Xie-like algorithm.
- The results are a little more accurate than earlier methods.

Extensions

- Assembly/Disassembly
- Buffer size optimization
- Closed loops
- Real-time control
- Quality/Quantity
- Multiple part types and reentrant flow
- Split/Merge
- Generalized two-machine, one-buffer line

Conclusions

- For some systems, randomness is an essential feature that defines the performance.
  - ⋆ For some systems you can approximate random variables by their averages, but for many you cannot.
- Randomness can be quantified and treated in practical ways.
- The appropriate approach to mitigate variability is proportionate to the time scale of the variability.
- We have a growing collection of practical tools for manufacturing system analysis, design, and operation.

Required Reading

http://home.comcast.net/~hierarchy/MSE/mse.html