Variability Basics

God does not play dice with the universe.

– Albert Einstein

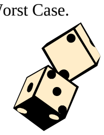
Stop telling God what to do.

– Niels Bohr

Variability Makes a Difference!

Little's Law: TH = WIP/CT, so same throughput can be obtained with large WIP, long CT or small WIP, short CT. The difference? ***Variability!***

Penny Fab One: achieves full TH (0.5 j/hr) at WIP=W_0=4 jobs if it behaves like Best Case, but requires WIP=27 jobs to achieve 95% of capacity if it behaves like the Practical Worst Case. Why? ***Variability!***



Tortise and Hare Example

Two machines:

- subject to same workload: 69 jobs/day (2.875 jobs/hr)
- subject to unpredictable outages (availability = 75%)

Hare X19:

- long, but infrequent outages

Tortoise 2000:

- short, but more frequent outages

Performance: Hare X19 is substantially worse on all measures than Tortoise 2000. Why?

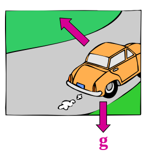
Variability!

Variability Views

Variability:
- Any departure from uniformity
- Random versus controllable variation

Randomness:
- Essential reality?
- Artifact of incomplete knowledge?
- Management implications: robustness is key

Probabilistic Intuition

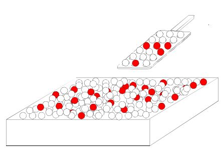


Uses of Intuition:
- driving a car
- throwing a ball
- mastering the stock market

First Moment Effects:
- Throughput increases with machine speed
- Throughput increases with availability
- Inventory increases with lot size
- Our intuition is good for first moments

Probabilistic Intuition (cont.)

Second Moment Effects:
- Which is more variable – processing times of parts or batches?
- Which are more disruptive – long, infrequent failures or short frequent ones?
- Our intuition is less secure for second moments
- Misinterpretation – e.g., regression to the mean



Variability

Definition: Variability is anything that causes the system to depart from regular, predictable behavior.

Sources of Variability:

- setups
- machine failures
- materials shortages
- yield loss
- rework
- operator unavailability
- workpace variation
- differential skill levels
- engineering change orders
- customer orders
- product differentiation
- material handling

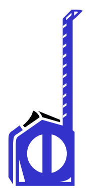
Measuring Process Variability

t_e = mean process time of a job

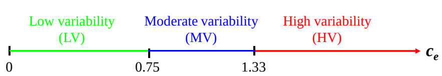
σ_e = standard deviation of process time

$c_e = \dfrac{\sigma_e}{t_e}$ = coefficient of variation, CV

Note: we often use the "squared coefficient of variation" (SCV), c_e^2



Variability Classes in Factory Physics®



Effective Process Times:

- ***actual*** process times are generally LV
- ***effective*** process times include setups, failure outages, etc.
- HV, LV, and MV are all possible in effective process times

Relation to Performance Cases: For balanced systems

- MV – Practical Worst Case
- LV – between Best Case and Practical Worst Case
- HV – between Practical Worst Case and Worst Case

Measuring Process Variability – Example

Trial	Machine 1	Machine 2	Machine 3
1	22	5	5
2	25	6	6
3	23	5	5
4	26	**35**	**35**
5	24	7	7
6	28	**45**	**45**
7	21	6	6
8	30	6	6
9	24	5	5
10	28	4	4
11	27	7	7
12	25	**50**	**500**
13	24	6	6
14	23	6	6
15	22	5	5
t_e	25.1	13.2	43.2
s_e	2.5	15.9	127.0
c_e	0.1	1.2	2.9
Class	LV	MV	HV

***Question*: can we measure c_e this way?**

***Answer*: No! Won't consider "rare" events properly.**

Natural Variability

Definition: variability without explicitly analyzed cause

Sources:

- operator pace
- material fluctuations
- product type (if not explicitly considered)
- product quality

Observation: natural process variability is usually in the LV category.

Down Time – Mean Effects

Definitions:

t_0 = base process time

c_0 = base process time coefficient of variability

$r_0 = \frac{1}{t_0}$ = base capacity (rate, e.g., parts/hr)

m_f = mean time to failure

m_r = mean time to repair

c_r = coefficent of variability of repair times (σ_r / m_r)

Down Time – Mean Effects (cont.)

Availability: Fraction of time machine is up

$$A = \frac{m_f}{m_f + m_r}$$

Effective Processing Time and Rate:

$$r_e = A r_0$$

$$t_e = t_0 / A$$

Totoise and Hare - Availability

Hare X19:

- t_0 = 15 min
- σ_0 = 3.35 min
- $c_0 = \sigma_0 / t_0$ = 3.35/15 = 0.05
- m_f = 12.4 hrs (744 min)
- m_r = 4.133 hrs (248 min)
- c_r = 1.0

Tortoise:

- t_0 = 15 min
- σ_0 = 3.35 min
- $c_0 = \sigma_0 / t_0$ = 3.35/15 = 0.05
- m_f = 1.9 hrs (114 min)
- m_r = 0.633 hrs (38 min)
- c_r = 1.0

Availability:

$$A = \frac{m_f}{m_f + m_r} = \frac{744}{744 + 248} = 0.75$$

$$A = \frac{m_f}{m_f + m_r} = \frac{114}{114 + 38} = 0.75$$

No difference between machines in terms of availability.

Down Time – Variability Effects

Effective Variability:

$$t_e = t_0 / A$$

$$\sigma_e^2 = \left(\frac{\sigma_0}{A}\right)^2 + \frac{(m_r^2 + \sigma_r^2)(1 - A) t_0}{A m_r}$$

Conclusions: $c_e^2 = \frac{\sigma_e^2}{t_e^2} = c_0^2 + (1 + c_r^2) A (1 - A) \frac{m_r}{t_0}$

Variability depends on repair times in addition to availability

- Failures inflate mean, variance, and CV of effective process time
- Mean (t_e) increases proportionally with $1/A$
- SCV (c_e^2) increases proportionally with m_r
- SCV (c_e^2) increases proportionally in c_r^2
- For constant availability (A), long infrequent outages increase SCV more than short frequent ones

Tortoise and Hare - Variability

Hare X19:

$t_e = \frac{t_0}{A} = \frac{15}{0.75} = 20 \text{ min}$

$c_e^2 = c_0^2 + (1+c_r^2)A(1-A)\frac{m_r}{t_0} =$

$0.05 + (1+1)0.75(1-0.75)\frac{248}{15} =$

6.25 high variability

Tortoise 2000

$t_e = \frac{t_0}{A} = \frac{15}{0.75} = 20 \text{ min}$

$c_e^2 = c_0^2 + (1+c_r^2)A(1-A)\frac{m_r}{t_0} =$

$0.05 + (1+1)0.75(1-0.75)\frac{38}{15} =$

1.0 moderate variability

Hare X19 is much more variable than Tortoise 2000!

Setups – Mean and Variability Effects

Analysis:

N_s = average no. jobs between setups

t_s = average setup duration

σ_s = std. dev. of setup time

$$c_s = \frac{\sigma_s}{t_s}$$

$$t_e = t_0 + \frac{t_s}{N_s}$$

$$\sigma_e^2 = \sigma_0^2 + \frac{\sigma_s^2}{N_s} + \frac{N_s - 1}{N_s^2} t_s^2$$

$$c_e^2 = \frac{\sigma_e^2}{t_e^2}$$

Setups – Mean and Variability Effects (cont.)

Observations:

- Setups increase mean *and* variance of processing times.
- Variability reduction is one benefit of flexible machines.
- However, the interaction is complex.

Setup – Example

Data:

- Fast, inflexible machine – 2 hr setup every 10 jobs

$$t_0 = 1 \text{ hr}$$
$$N_s = 10 \text{ jobs/setup}$$
$$t_s = 2 \text{ hrs}$$
$$t_e = t_0 + t_s / N_s = 1 + 2/10 = 1.2 \text{ hrs}$$
$$r_e = 1/t_e = 1/(1+2/10) = 0.8333 \text{ jobs/hr}$$

- Slower, flexible machine – no setups

$$t_0 = 1.2 \text{ hrs}$$
$$r_e = 1/t_0 = 1/1.2 = 0.833 \text{ jobs/hr}$$

Traditional Analysis?

No difference!

Setup – Example (cont.)

Factory Physics® Approach: Compare mean *and* variance

- Fast, inflexible machine – 2 hr setup every 10 jobs

$$t_0 = 1 \text{ hr}$$
$$c_0^2 = 0.0625$$
$$N_s = 10 \text{ jobs/setup}$$
$$t_s = 2 \text{ hrs}$$
$$c_s^2 = 0.0625$$
$$t_e = t_0 + t_s / N_s = 1 + 2/10 = 1.2 \text{ hrs}$$
$$r_e = 1/t_e = 1/(1+2/10) = 0.8333 \text{ jobs/hr}$$
$$\sigma_e^2 = \sigma_0^2 + t_s^2 \left(\frac{c_s^2}{N_s} + \frac{N_s - 1}{N_s^2} \right) = 0.4475$$
$$c_e^2 = 0.31$$

Setup – Example (cont.)

- Slower, flexible machine – no setups

$$t_0 = 1.2 \text{ hrs}$$
$$c_0^2 = 0.25$$

$$r_e = 1/t_0 = 1/1.2 = 0.833 \text{ jobs/hr}$$
$$c_e^2 = c_0^2 = 0.25$$

Conclusion:

Flexibility can reduce variability.

Setup – Example (cont.)

New Machine: Consider a third machine same as previous machine with setups, but with shorter, more frequent setups

$N_s = 5$ jobs/setup

$t_s = 1$ hr

Analysis:

$$r_e = 1/t_e = 1/(1+1/5) = 0.833 \text{ jobs/hr}$$

$$\sigma_e^2 = \sigma_0^2 + t_s^2\left(\frac{c_s^2}{N_s} + \frac{N_s - 1}{N_s^2}\right) = 0.2350$$

$$c_e^2 = 0.16$$

Conclusion: ***Shorter, more frequent setups induce less variability.***

Other Process Variability Inflators

Sources:

- operator unavailability
- recycle
- batching
- material unavailability
- et cetera, et cetera, et cetera

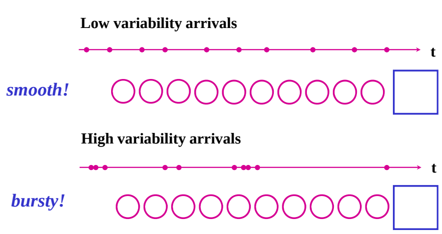
Effects:

- inflate t_e
- inflate c_e

Consequences:

Effective process variability can be LV, MV,or HV.

Illustrating Flow Variability



Measuring Flow Variability

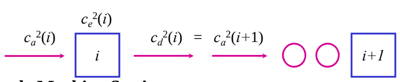
t_a = mean time between arrivals

$r_a = \frac{1}{t_a}$ = arrival rate

σ_a = standard deviation of time between arrivals

$c_a = \frac{\sigma_a}{t_a}$ = coefficient of variation of interarrival times

Propagation of Variability



Single Machine Station:

$$\boxed{c_d^2 = u^2 c_e^2 + (1-u^2) c_a^2}$$

where u is the station utilization given by $u = r_a t_e$

departure var depends on arrival var and process var

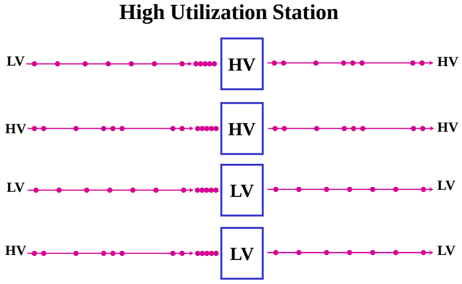
Multi-Machine Station:

$$c_d^2 = 1 + (1-u^2)(c_a^2 - 1) + \frac{u^2}{\sqrt{m}}(c_e^2 - 1)$$

where m is the number of (identical) machines and

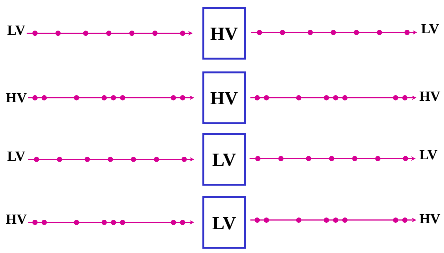
$$u = \frac{r_a t_e}{m}$$

Propagation of Variability – High Utilization Station



***Conclusion**: flow variability out of a high utilization station is determined primarily by process variability at that station.*

Propagation of Variability – Low Utilization Station



***Conclusion**: flow variability out of a low utilization station is determined primarily by flow variability into that station.*

Variability Interactions

Importance of Queueing:

- manufacturing plants are ***queueing networks***
- queueing and waiting time comprise majority of cycle time

System Characteristics:

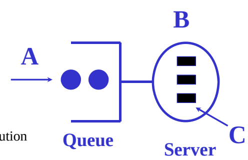
- Arrival process
- Service process
- Number of servers
- Maximum queue size (blocking)
- Service discipline (FCFS, LCFS, EDD, SPT, etc.)
- Balking
- Routing
- Many more

Kendall's Classification

A/B/C

A: arrival process
B: service process
C: number of machines

M: exponential (Markovian) distribution
G: completely general distribution
D: constant (deterministic) distribution.



Queueing Parameters

r_a = the rate of arrivals in customers (jobs) per unit time ($t_a = 1/r_a$ = the average time between arrivals).

c_a = the CV of inter-arrival times.

m = the number of machines.

r_e = the rate of the station in jobs per unit time = m/t_e.

c_e = the CV of ***effective*** process times.

Note: a station can be described with 5 parameters.

u = utilization of station = r_a/r_e.

Queueing Measures

Measures:

- CT_q = the expected waiting time spent in queue.
- CT = the expected time spent at the process center, i.e., queue time plus process time.
- WIP = the average WIP level (in jobs) at the station.
- WIP_q = the expected WIP (in jobs) in queue.

Relationships:

$CT = CT_q + t_e$
$WIP = r_a \times CT$
$WIP_q = r_a \times CT_q$

Result: If we know CT_q, we can compute WIP, WIP_q, CT.

The G/G/1 Queue

Formula:

$$\boxed{\begin{aligned} CT_q &\approx V \times U \times t \\ &\approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e \end{aligned}}$$

Observations:

- Useful model of single machine workstations
- Separate terms for variability, utilization, process time.
- CT_q (and other measures) increase with c_a^2 and c_e^2
- Flow variability, process variability, or both can combine to inflate queue time.
- ***Variability causes congestion!***

The G/G/m Queue

Formula:

$$\mathrm{CT}_q \approx V \times U \times t$$
$$\approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u^{\sqrt{2(m+1)}-1}}{m(1-u)}\right)t_e$$

Observations:

- Useful model of multi-machine workstations
- *Extremely* general.
- Fast and accurate.
- Easily implemented in a spreadsheet (or packages like MPX).

VUT Spreadsheet

	MEASURE:	STATION:	1	2	3	4	5
basic data	Arrival Rate (parts/hr)	r_a	10.000	9.800	9.310	8.845	7.960
	Arrival CV	c_a^2	1.000	0.181	0.031	0.061	0.035
	Natural Process Time (hr)	t_0	0.090	0.090	0.095	0.090	0.090
	Natural Process SCV	c_0^2	0.500	0.500	0.500	0.500	0.500
	Number of Machines	m	1	1	1	1	1
failures	MTTF (hr)	m_f	200	200	200	200	200
	MTTR (hr)	m_r	2	2	8	4	4
	Availability	A	0.990	0.990	0.962	0.980	0.980
	Effective Process Time (failures only)	t_e'	0.091	0.091	0.099	0.092	0.092
	Eff Process SCV (failures only)	$c_e^{2\prime}$	0.936	0.936	6.729	2.209	2.209
setups	Batch Size	k	100	100	100	100	100
	Setup Time (hr)	t_s	0.000	0.500	0.500	0.000	0.000
	Setup Time SCV	c_s^2	1.000	1.000	1.000	1.000	1.000
	Arrival Rate of Batches	r_a/k	0.100	0.098	0.093	0.088	0.080
	Eff Batch Process Time (failures+setups)	$t_e = kt_0/A+t_s$	9.090	9.590	10.380	9.180	9.180
	Eff Batch Process Time Var (failures+setups)	$k*\sigma_0^2/A^2+2m_r(1-A)kt_0/A+\sigma_s^2$	0.773	1.023	6.818	1.861	1.861
	Eff Process SCV (failures+setups)	c_e^2	0.009	0.011	0.063	0.022	0.022
	Utilization	u	0.909	0.940	0.966	0.812	0.731
	Departure SCV	c_d^2	0.181	0.031	0.061	0.035	0.028
yield	Yield	y	0.980	0.950	0.950	0.900	0.950
	Final Departure Rate	r_a*y	9.800	9.310	8.845	7.960	7.562
	Final Departure SCV	$yc_d^2+(1-y)$	0.198	0.079	0.108	0.132	0.077
measures	Utilization	u	0.909	0.940	0.966	0.812	0.731
	Throughput	TH	9.800	9.310	8.845	7.960	7.562
	Queue Time (hr)	CT_q	45.825	14.421	14.065	1.649	0.716
	Cycle Time (hr)	CT_q+t_e	54.915	24.011	24.445	10.829	9.896
	Cumulative Cycle Time (hr)	$\Sigma_i(CT_q(i)+t_e(i))$	54.915	78.925	103.371	114.200	124.096
	WIP in Queue (jobs)	r_aCT_q	458.249	141.321	130.948	14.587	5.700
	WIP (jobs)	r_aCT	549.149	235.303	227.586	95.780	78.773
	Cumulative WIP (jobs)	$\Sigma_i(r_a(i)CT(i))$	549.149	784.452	1012.038	1107.818	1186.591

Effects of Blocking

VUT Equation:

- characterizes stations with infinite space for queueing
- useful for seeing what will happen to WIP, CT without restrictions

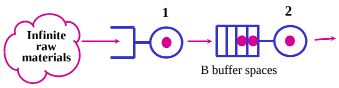
But real world systems often constrain WIP:

- physical constraints (e.g., space or spoilage)
- logical constraints (e.g., kanbans)

Blocking Models:

- estimate WIP and TH for given set of rates, buffer sizes
- much more complex than non-blocking (open) models, often require simulation to evaluate realistic systems

The M/M/1/b Queue



Note: ***there is room for b=B+2 jobs in system, B in the buffer and one at each station.***

Model of Station 2

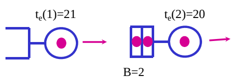
$$WIP(M/M/1/b) = \frac{u}{1-u} - \frac{(b+1)u^{b+1}}{1-u^{b+1}}$$ ← ***Goes to u/(1-u) as b→∞ Always less than WIP(M/M/1)***

$$TH(M/M/1/b) = \frac{1-u^b}{1-u^{b+1}} r_a$$ ← ***Goes to r_a as b→∞ Always less than TH(M/M/1)***

$$CT(M/M/1/b) = \frac{WIP(M/M/1/b)}{TH(M/M/1/b)}$$ ***Little's law***

where $u = t_e(2)/t_e(1)$ ***Note:*** ***u>1 is possible; formulas valid for u≠1***

Blocking Example



$$u = t_e(2)/t_e(1) = 20/21 = 0.9524$$

$$WIP(M/M/1) = \frac{u}{1-u} = 20 \text{ jobs}$$

$$TH(M/M/1) = r_a = 1/t_e(1) = 1/21 = 0.0476 \text{ job/min}$$

$$TH(M/M/1/b) = \frac{1-u^b}{1-u^{b+1}} r_a = \frac{1-0.9524^4}{1-0.9524^5}\left(\frac{1}{21}\right) = 0.039 \text{ job/min}$$ ***18% less TH***

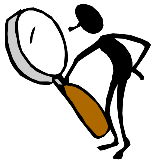
$$WIP(M/M/1/b) = \frac{u}{1-u} - \frac{(b+1)u^{b+1}}{1-u^{b+1}} = 20 - \frac{5(0.9524^5)}{1-0.9524^5} = 1.8954 \text{ jobs}$$ ***90% less WIP***

M/M/1/b system has less WIP and less TH than M/M/1 system

Seeking Out Variability

General Strategies:

- look for long queues (Little's law)
- look for blocking
- focus on high utilization resources
- consider both flow and process variability
- ask "why" five times



Specific Targets:

- equipment failures
- setups
- rework
- operator pacing
- anything that prevents regular arrivals and process times

Variability Pooling

Basic Idea: the CV of a sum of independent random variables decreases with the number of random variables.

Example (Time to process a batch of parts):

$$t_0 = \text{time to process single part}$$
$$\sigma_0 = \text{standard deviation of time to process single part}$$
$$c_0 = \frac{\sigma_0}{t_0} = \text{CV of time to process single part}$$
$$t_0(batch) = nt_0$$
$$\sigma_0^2(batch) = n\sigma_0^2$$
$$c_0^2(batch) = \frac{\sigma_0^2(batch)}{t_0^2(batch)} = \frac{n\sigma_0^2}{n^2t_0^2} = \frac{\sigma_0^2}{nt_0^2} = \frac{c_0^2}{n} \Rightarrow \boxed{c_0(batch) = \frac{c_0}{\sqrt{n}}}$$

Safety Stock Pooling Example

- **PC's consist of 6 components (CPU, HD, CD ROM, RAM, removable storage device, keyboard)**
- **3 choices of each component: 3^6 = 729 different PC's**
- **Each component costs \$150 (\$900 material cost per PC)**
- **Demand for all models is normally distributed with mean 100 per year, standard deviation 10 per year**
- **Replenishment lead time is 3 months, so average demand during LT is $\theta = 25$ for computers and $\theta = 25(729/3) = 6075$ for components**
- **Use base stock policy with fill rate of 99%**

Pooling Example - Stock PC's

Base Stock Level for Each PC:

cycle stock *safety stock*

$$R = \theta + z_s\sigma = 25 + 2.33(\sqrt{25}) = 37$$

On-Hand Inventory for Each PC:

$$I(R) = R - \theta + B(R) \approx R - \theta = z_s\sigma = 37 - 25 = 12 \text{ units}$$

Total (Approximate) On-Hand Inventory :

$$12 \times 729 \times \$900 = \$7,873,200$$

Pooling Example - Stock Components

Necessary Service for Each Component:

$S = (0.99)^{1/6} = 0.9983 \Rightarrow z_s = 2.93$

Base Stock Level for Each Component:

$R = \theta + z_s\sigma = \underbrace{6075}_{\text{cycle stock}} + \underbrace{2.93(\sqrt{6075})}_{\text{safety stock}} = 6303$

On-Hand Inventory Level for Each Component:

$I(R) = R - \theta + B(R) \approx R - \theta = z_s\sigma = 6303\text{-}6075 = 228 \text{ units}$

Total Safety Stock:

$228 \times 18 \times \$150 = \$615{,}600$ ***92% reduction!***

Basic Variability Takeaways

Variability Measures:

- CV of effective process times
- CV of interarrival times

Components of Process Variability

- failures
- setups
- many others - deflate capacity *and* inflate variability
- long infrequent disruptions worse than short frequent ones

Consequences of Variability:

- variability causes congestion (i.e., WIP/CT inflation)
- variability propagates
- variability and utilization interact
- pooled variability less destructive than individual variability

The Corrupting Influence of Variability

When luck is on your side, you can do without brains.

– Giordano Bruno,burned
at the stake in 1600

The more you know the luckier you get.

– "J.R. Ewing" of *Dallas*

Performance of a Serial Line

Measures:
- Throughput
- Inventory (RMI, WIP, FGI)
- Cycle Time
- Lead Time
- Customer Service
- Quality

Evaluation:
- Comparison to "perfect" values (e.g., r_b, T_0)
- Relative weights consistent with business strategy?

Links to Business Strategy:
- Would inventory reduction result in significant cost savings?
- Would CT (or LT) reduction result in significant competitive advantage?
- Would TH increase help generate significantly more revenue?
- Would improved customer service generate business over the long run?

Remember – standards change over time!

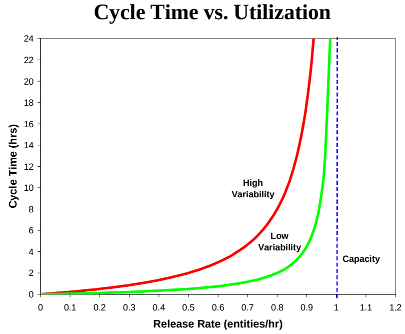
Capacity Laws

Capacity Law: ***In steady state, all plants will release work at an average rate that is strictly less than average capacity.***

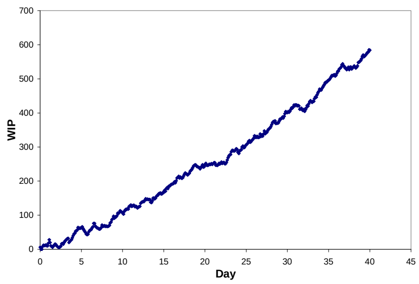
Utilization Law: ***If a station increases utilization without making any other change, average WIP and cycle time will increase in a highly nonlinear fashion.***

Notes:
- Cannot run at full capacity (including overtime, etc.)
- Failure to recognize this leads to "fire fighting"

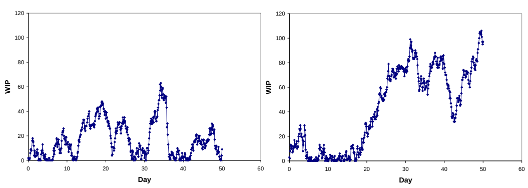
Cycle Time vs. Utilization



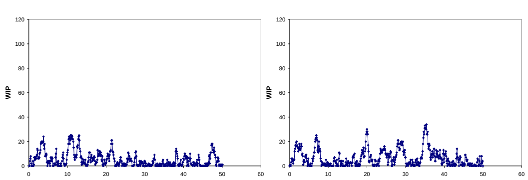
What Really Happens: System with Insufficient Capacity



What Really Happens: Two Cases with Releases at 100% of Capacity



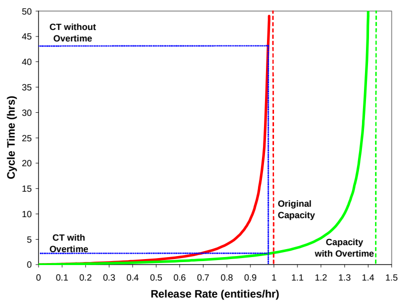
What Really Happens: Two Cases with Releases at 82% of Capacity



Overtime Vicious Cycle

1. **Release work at plant capacity.**
2. **Variability causes WIP to increase.**
3. **Jobs are late, customers complain,…**
4. **Authorize one-time use of overtime.**
5. **WIP falls, cycle times go down, backlog is reduced.**
6. **Breathe sigh of relief.**
7. **Go to Step 1!**

Mechanics of Overtime Vicious Cycle



Influence of Variability

Variability Law: ***Increasing variability always degrades the performance of a production system.***

Examples:

- process time variability pushes best case toward worst case
- higher demand variability requires more safety stock for same level of customer service
- higher cycle time variability requires longer lead time quotes to attain same level of on-time delivery

Variability Buffering

Buffering Law: ***Systems with variability must be buffered by some combination of:***

1. inventory
2. capacity
3. time.

Interpretation: If you cannot pay to reduce variability, you will pay in terms of high WIP, under-utilized capacity, or reduced customer service (i.e., lost sales, long lead times, and/or late deliveries).

Variability Buffering Examples

Ballpoint Pens:

- can't buffer with time (who will backorder a cheap pen?)
- can't buffer with capacity (too expensive, and slow)
- must buffer with inventory

Ambulance Service:

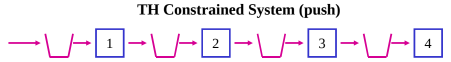
- can't buffer with inventory (stock of emergency services?)
- can't buffer with time (violates strategic objectives)
- must buffer with capacity

Organ Transplants:

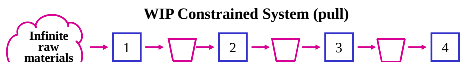
- can't buffer with WIP (perishable)
- can't buffer with capacity (ethically anyway)
- must buffer with time

Simulation Studies

TH Constrained System (push)



WIP Constrained System (pull)



r_a = arrival rate
c_a = CV of interarrival times
$t_e(i)$ = effective process time at station i
$c_e(i)$ = effective CV at station i
$B(i)$ = buffer size in front of station i

Variability in Push Systems

Case	$t_e(i)$, i = 1, 2, 4 (min)	$t_e(3)$ (min)	$c(i)$, i = 1-4 (unitless)	TH (j/min)	CT (min)	WIP (jobs)	σ_{CT} (min)	Comments
1	1	1.2	0	0.8	4.2	3.4	0.0	***best case***
2	1	1.2	1	0.8	44.6	35.7	26.8	***WIP buffer***
3	1	1.0	1	0.8	20.0	16.0	10.3	***capacity buffer***
4	1	1.2	0.3	0.8	7.8	6.2	3.3	***reduced variability***

Notes:

- $r_a = 0.8$, $c_a = c_e(i)$ in all cases.
- $B(i) = \infty$, i = 1-4 in all cases.

Observations:

- TH is set by release rate in a push system.
- Increasing capacity (r_b) reduces need for WIP buffering.
- Reducing process variability reduces WIP, CT, and CT variability for a given throughput level.

Variability in Pull Systems

Case	$t_e(i)$, i = 1,2,4 (min)	$t_e(3)$ (min)	$c(i)$, i = 1-4 (unitless)	$B(3)$ (jobs)	TH (j/min)	CT (min)	WIP (jobs)	σ_{CT} (min)	Comments
1	1	1.2	0	0	0.83	4.6	3.8	0.0	***best case***
2	1	1.2	1	0	0.48	6.4	3.1	2.4	***plain JIT***
3	1	1.2	1	1	0.53	7.2	3.8	2.6	***inv buffer***
4	1	1.2	0.3	0	0.72	5.0	3.6	0.6	***var reduction***
5	1	1.2	0.3	1	0.76	6.0	4.5	0.8	***inv buffer + var reduction***
6	1	1.2	0.3	0	0.73	6.3	4.6	0.7	***non-bottleneck buffer***

Notes:

- Station 1 pulls in job whenever it becomes empty.
- $B(i) = 0$, i = 1, 2, 4 in all cases, except case 6, which has $B(2) = 1$.

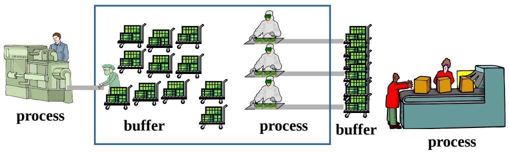
Variability in Pull Systems (cont.)

Observations:

- Capping WIP without reducing variability reduces TH.
- WIP cap limits effect of process variability on WIP/CT.
- Reducing process variability increases TH, given same buffers.
- Adding buffer space at bottleneck increases TH.
- Magnitude of impact of adding buffers depends on variability.
- Buffering less helpful at non-bottlenecks.
- Reducing process variability reduces CT variability.

Conclusion: ***consequences of variability are different in push and pull systems, but in either case the buffering law implies that you will pay for variability somehow.***

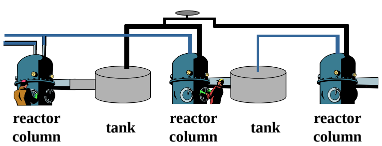
Example – Discrete Parts Flowline



Inventory Buffers: raw materials, WIP between processes, FGI
Capacity Buffers: overtime, equipment capacity, staffing
Time Buffers: frozen zone, time fences, lead time quotes

Variability Reduction: smaller WIP & FGI , shorter cycle times

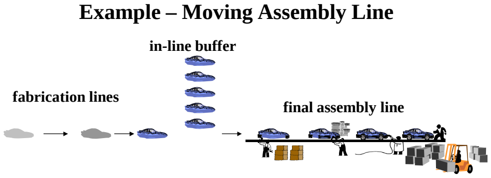
Example – Batch Chemical Process



Inventory Buffers: raw materials, WIP in tanks, finished goods
Capacity Buffers: idle time at reactors
Time Buffers: lead times in supply chain

Variability Reduction: WIP is tightly constrained, so target is primarily throughput improvement, and maybe FGI reduction.

Example – Moving Assembly Line



Inventory Buffers: components, in-line buffers
Capacity Buffers: overtime, rework loops, warranty repairs
Time Buffers: lead time quotes

Variability Reduction: initially directed at WIP reduction, but later to achieve better use of capacity (e.g., more throughput)

Buffer Flexibility

Buffer Flexibility Corollary: ***Flexibility reduces the amount of variability buffering required in a production system.***

Examples:

- Flexible Capacity: cross-trained workers
- Flexible Inventory: generic stock (e.g., assemble to order)
- Flexible Time: variable lead time quotes

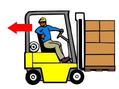
Variability from Batching

VUT Equation:

- CT depends on process variability *and* flow variability

Batching:

- affects flow variability
- affects waiting inventory



Conclusion: batching is an important determinant of performance

Process Batch Versus Move Batch

Dedicated Assembly Line: ***What should the batch size be?***

Process Batch:

- Related to length of *setup*.
- The longer the setup the larger the lot size required for the same capacity.

Move (transfer) Batch: ***Why should it equal process batch?***

- The smaller the move batch, the shorter the cycle time.
- The smaller the move batch, the more material handling.

Lot Splitting: ***Move batch can be different from process batch.***

1. Establish smallest economical move batch.
2. Group batches of like families together at bottleneck to avoid setups.
3. Implement using a "backlog".

Process Batching Effects

Types of Process Batching:

1. ***Serial Batching***:
   - processes with sequence-dependent setups
   - "batch size" is number of jobs between setups
   - batching used to reduce loss of capacity from setups

2. ***Parallel Batching***:
   - true "batch" operations (e.g., heat treat)
   - "batch size" is number of jobs run together
   - batching used to increase effective rate of process

Process Batching

Process Batching Law: ***In stations with batch operations or significant changeover times:***

1. ***The minimum process batch size that yields a stable system may be greater than one.***
2. ***As process batch size becomes large, cycle time grows proportionally with batch size.***
3. ***Cycle time at the station will be minimized for some process batch size, which may be greater than one.***

Basic Batching Tradeoff: WIP versus capacity

Serial Batching

Parameters:

k = serial batch size (10)

t = time to process a single part (1)

s = time to perform a setup (5)

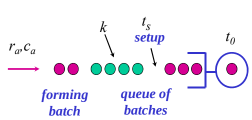
c_e = CV for batch (parts + setup) (0.5)

r_a = arrival rate for parts (0.4)

c_a = CV of batch arrivals (1.0)

Time to process batch: $t_e = kt + s$

$t_e = 10(1) + 5 = 15$



Process Batching Effects (cont.)

Arrival rate of batches: r_a/k

$r_a = 0.4/10 = 0.04$

Utilization: $u = (r_a/k)(kt + s)$

$u = 0.04(10 \cdot 1+5) = 0.6$

For stability: $u < 1$ requires $k > \dfrac{sr_a}{1 - tr_a}$

minimum batch size required for stability of system...

$$k > \frac{5(0.4)}{1 - 1(0.4)} = 3.33$$

Process Batching Effects (cont.)

Average queue time at station:

$$\mathrm{CT}_q = \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e = \left(\frac{1+0.5}{2}\right)\left(\frac{0.6}{1-0.6}\right)15 = 16.875$$

Note: we assume arrival CV of batches is c_a regardless of batch size – an approximation...

Average cycle time depends on move batch size:

- Move batch = process batch

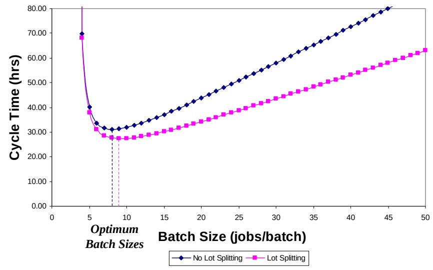
$$\mathrm{CT}_{\text{non-split}} = \mathrm{CT}_q + t_e = \mathrm{CT}_q + s + kt$$
$$= 16.875 + 15 = 31.875$$

- Move batch = 1

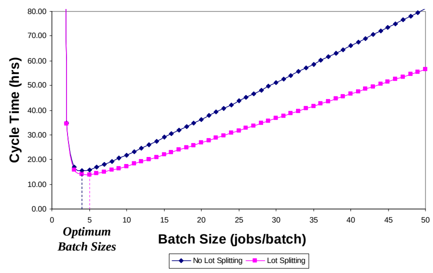
$$\mathrm{CT}_{\text{split}} = \mathrm{CT}_q + s + \frac{k+1}{2}t$$
$$= 16.875 + 10 + \frac{10+1}{2}(1.0) = 27.375$$

Note: splitting move batches reduces wait for batch time.

Cycle Time vs. Batch Size – 5 hr setup



Cycle Time vs. Batch Size – 2.5 hr setup



Setup Time Reduction

Where?

- Stations where capacity is expensive
- Excess capacity may sometimes be cheaper

Steps:

1. Externalize portions of setup
2. Reduce adjustment time (guides, clamps, etc.)
3. Technological advancements (hoists, quick-release, etc.)

Caveat: Don't count on capacity increase; more flexibility will require more setups.

Parallel Batching

Parameters:

k = parallel batch size (10)

t = time to process a batch (90)

c_e = CV for batch (1.0)

r_a = arrival rate for parts (0.05)

c_a = CV of batch arrivals (1.0)

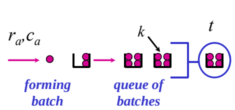
B = maximum batch size (100)

Time to form batch: $W = \frac{k-1}{2}\frac{1}{r_a}$

W = ((10 – 1)/2)(1/0.005) = 90

Time to process batch: $t_e = t$

t_e = 90



Parallel Batching (cont.)

Arrival of batches: r_a/k

$r_a/k = 0.05/10 = 0.005$

Utilization: $u = (r_a/k)(t)$

$u = (0.005)(90) = 0.45$

For stability: u < 1 requires $k > r_a t$ ***minimum batch size required for stability of system...***

$k > 0.05(90) = 4.5$

Parallel Batching (cont.)

Average wait-for-batch time:

$$\text{WT} = \frac{k-1}{2}\frac{1}{r_a} = \frac{10-1}{2}\frac{1}{0.05} = 90$$

batch size affects both wait-for-batch time and queue time

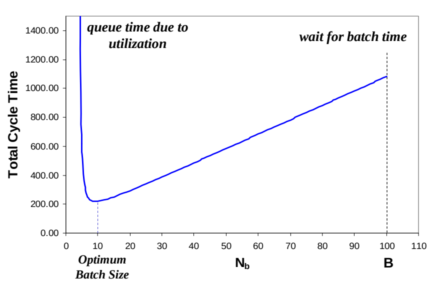
Average queue plus process time at station:

$$\text{CT} = \left(\frac{c_a^2/k + c_0^2}{2}\right)\left(\frac{u}{1-u}\right)t + t = \left(\frac{0.1+1}{2}\right)\left(\frac{0.45}{1-0.45}\right)90 + 90 = 130.5$$

Total cycle time:

$$\text{CT} + \text{WT} = 90 + 130.5 = 220.5$$

Cycle Time vs. Batch Size in a Parallel Operation



Variable Batch Sizes

Observation: Waiting for full batch in parallel batch operation may not make sense. Could just process whatever is there when operation becomes available.

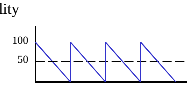
Example:

- Furnace has space for 120 wrenches
- Heat treat requires 1 hour
- Demand averages 100 wrenches/hr
- Induction coil can heat treat 1 wrench in 30 seconds
- What is difference between performance of furnace and coil?

Variable Batch Sizes (cont.)

Furnace: Ignoring queueing due to variability

- Process starts every hour
- 100 wrenches in furnace
- 50 wrenches waiting on average
- 150 total wrenches in WIP
- CT = WIP/TH = 150/100 = 3/2 hr = 90 min



Induction Coil: Capacity same as furnace (120 wrenches/hr), but

- CT = 0.5 min = 0.0083 hr
- WIP = TH × CT = 100 × 0.0083 = 0.83 wrenches

Conclusion: Dramatic reduction in WIP and CT due to small batches—independent of variability or other factors.

Move Batching

<u>Move Batching Law</u>: Cycle times over a segment of a routing are roughly proportional to the transfer batch sizes used over that segment, provided there is no waiting for the conveyance device.

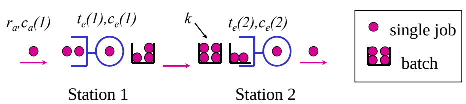
Insights:

- Basic Batching Tradeoff: WIP vs. move frequency
- Queueing for conveyance device can offset CT reduction from reduced move batch size
- Move batching intimately related to material handling and layout decisions

Move Batching

Problem:

- Two machines in series
- First machine receives individual parts at rate r_a with CV of $c_a(1)$ and puts out batches of size k.
- First machine has mean process time of $t_e(1)$ for one part with CV of $c_e(1)$.
- Second machine receives batches of k and put out individual parts.
- How does cycle time depend on the batch size k?



Move Batching Calculations

Time at First Station:

- Average time before batching is:

$$\frac{c_a^2(1)+c_e^2(1)}{2}\frac{u(1)}{1-u(1)}t_e(1)+t_e(1)$$

regular VUT equation...

- Average time forming the batch is:

$$\frac{k-1}{2}\frac{1}{r_a}=\frac{k-1}{2u(1)}t_e(1)$$

first part waits (k-1)(1/r_a), last part doesn't wait, so average is (k-1)(1/r_a)/2

- Average time spent at the first station is:

$$\mathrm{CT}(1)=\frac{c_a^2(1)+c_e^2(1)}{2}\frac{u(1)}{1-u(1)}t_e(1)+t_e(1)+\frac{k-1}{2u(1)}t_e(1)$$

$$=\mathrm{CT}(1,\text{no batching})+\frac{k-1}{2u(1)}t_e(1)$$

Move Batching Calculations (cont.)

Output of First Station:

- Time between output of individual parts into the batch is t_a.
- Time between output of batches of size k is kt_a.
- Variance of interoutput times of parts is $c_d^2(1)t_a^2$, where

$$c_d^2(1)=(1-u(1)^2)c_a^2(1)+u(1)^2c_e^2(1)$$

because $c_d^2(1)=\sigma_d^2/t_a^2$ by def of CV

- Variance of batches of size k is $kc_d^2(1)t_a^2$.
- SCV of batch arrivals to station 2 is:

because departures are independent, so variances add

$$c_a^2(2)=\frac{kc_d^2(1)t_a^2}{k^2t_a^2}=\frac{c_d^2(1)}{k}$$

variance divided by mean squared...

Move Batching Calculations (cont.)

Time at Second Station:

independent process times...

- Time to process a batch of size k is $kt_e(2)$.
- Variance of time to process a batch of size k is $kc_e^2(2)t_e^2(2)$.
- SCV for a batch of size k is:

$$\frac{kc_e^2(2)t_e^2(2)}{k^2t_e^2(2)} = \frac{c_e^2(2)}{k}$$

- Mean time spent in partial batch of size k is:

$$\frac{k-1}{2}t_e(2)$$

first part doesn't wait, last part waits (k-1)t_e(2), so average is (k-1)t_e(2)/2

- So, average time spent at the second station is:

$$\text{CT}(2) = \frac{c_d^2(1)/k + c_e^2(2)/k}{2}\,\frac{u(2)}{1-u(2)}\,kt_e(2) + \frac{k-1}{2}t_e(2) + t_e(2)$$

$$= \text{CT}(2, \text{no batching}) + \frac{k-1}{2}t_e(2)$$

VUT equation to compute queue time of batches...

Move Batching Calculations (cont.)

Total Cycle Time:

$$\text{CT(batching)} = \text{CT(no batching)} + \frac{k-1}{2u(1)}t_e(1) + \frac{k-1}{2}t_e(2)$$

$$= \text{CT(no batching)} + \underbrace{\left(\frac{k-1}{2}\right)\left(\frac{t_e(1)}{u(1)} + t_e(2)\right)}_{\text{inflation factor due to move batching}}$$

Insight:

- Cycle time increases with k.
- Inflation term does not involve CV's
- Congestion from batching is more ***bad control*** than ***randomness***.

Assembly Operations

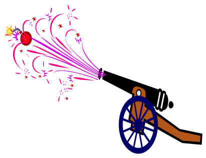
Assembly Operations Law: ***The performance of an assembly station is degraded by increasing any of the following:***

1. ***Number of components being assembled.***
2. ***Variability of component arrivals.***
3. ***Lack of coordination between component arrivals.***

Observations:

- This law can be viewed as special instance of variability law.
- Number of components affected by product/process design.
- Arrival variability affected by process variability and production control.
- Coordination affected by scheduling and shop floor control.

Attacking Variability



Objectives

- reduce cycle time
- increase throughput
- improve customer service

Levers

- reduce variability directly
- buffer using inventory
- buffer using capacity
- buffer using time
- increase buffer flexibility

Cycle Time

Definition (Station Cycle Time): ***The average cycle time at a station is made up of the following components:***

cycle time = move time + queue time + setup time + process time + wait-to-batch time + wait-in-batch time + wait-to-match time

delay times typically make up 90% of CT

Definition (Line Cycle Time): ***The average cycle time in a line is equal to the sum of the cycle times at the individual stations less any time that overlaps two or more stations.***

Reducing Queue Delay

$$CT_q = V \times U \times t$$

$$\left(\frac{c_a^2 + c_e^2}{2}\right) \qquad \left(\frac{u}{1-u}\right)$$

Reduce Variability
- failures
- setups
- uneven arrivals, etc.

Reduce Utilization
- arrival rate (yield, rework, etc.)
- process rate (speed, time, availability, etc)

Reducing Batching Delay

CT_{batch} = *delay at stations + delay between stations*

Reduce Process Batching
- Optimize batch sizes
- Reduce setups
  - Stations where capacity is expensive
  - Capacity vs. WIP/CT tradeoff

Reduce Move Batching
- Move more frequently
- Layout to support material handling (e.g., cells)

Reducing Matching Delay

CT_{batch} = *delay due to lack of synchronization*

Reduce Variability
- on high utilization fabrication lines
- usual variability reduction methods

Improve Coordination
- scheduling
- pull mechanisms
- modular designs

Reduce Number of Components
- product redesign
- kitting

Increasing Throughput

$TH = P(\text{bottleneck is busy}) \times \text{bottleneck rate}$

Reduce Blocking/Starving
- buffer with inventory (near bottleneck)
- reduce system "desire to queue"

$CT_q = V \times U \times t$

Reduce Variability

Reduce Utilization

Increase Capacity
- add equipment
- increase operating time (e.g. spell breaks)
- increase reliability
- reduce yield loss/rework

Note*: *if WIP is limited, then system degrades via TH loss rather than WIP/CT inflation

Customer Service

Elements of Customer Service:

- lead time
- fill rate (% of orders delivered on-time)
- quality

Law (Lead Time): ***The manufacturing lead time for a routing that yields a given service level is an increasing function of both the mean and standard deviation of the cycle time of the routing.***

Improving Customer Service

$$LT = CT + z\,\sigma_{CT}$$

Reduce CT Visible to Customer

- delayed differentiation
- assemble to order
- stock components

Reduce Average CT

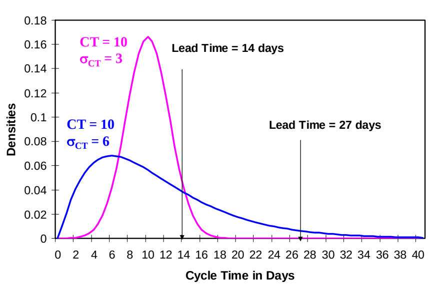
- queue time
- batch time
- match time

Reduce CT Variability

generally same as methods for reducing average CT:

- improve reliability
- improve maintainability
- reduce labor variability
- improve quality
- improve scheduling, etc.

Cycle Time and Lead Time



Diagnostics Using Factory Physics®

Situation:

- Two machines in series; machine 2 is bottleneck
- $c_a^2 = 1$
- Machine 1:
  $$t_0 = 19\,\text{min}$$
  $$c_0^2 = 0.25$$
  $$\text{MTTF} = 48\,\text{hr},\ \text{MTTR} = 8\,\text{hr}$$
- Machine 2:
  $$t_0 = 22\,\text{min}$$
  $$c_0^2 = 1$$
  $$\text{MTTF} = 3.3\,\text{hr},\ \text{MTTR} = 10\,\text{min}$$
  - Space at machine 2 for 20 jobs of WIP
- Desired throughput 2.4 jobs/hr, not being met

Diagnostic Example (cont.)

Proposal: Install second machine at station 2

- Expensive
- Very little space

Analysis Tools:

$$\text{CT}_q \approx \frac{c_a^2 + c_e^2}{2}\frac{u}{1-u}t_e$$ *VUT equation*

$$c_d^2 = u^2 c_e^2 + (1-u^2)c_a^2$$ *propogation equation*

Analysis:

Ask why five times...

Step 1: At 2.4 job/hr

- CT_q at first station is 645 minutes, average WIP is 25.8 jobs.
- CT_q at second station is 892 minutes, average WIP is 35.7 jobs.
- Space requirements at machine 2 are violated!

Diagnostic Example (cont.)

Step 2: **Why** is CT_q at machine 2 so big?

- Break CT_q into
  $$\text{CT}_q \approx \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e = (3.16)(12.22)(23.11\,\text{min})$$
- The 23.11 min term is small.
- The 12.22 correction term is moderate ($u \approx 0.9244$)
- The 3.16 correction is large.

Step 3: **Why** is the correction term so large?

- Look at components of correction term.
- $c_e^2 = 1.04$, $c_a^2 = 5.27$.
- Arrivals to machine are highly variable.

Diagnostic Example (cont.)

Step 4: **Why** is c_a^2 to machine 2 so large?

- Recall that c_a^2 to machine 2 equals c_d^2 from machine 1, and
  $$c_d^2 = u^2 c_e^2 + (1-u^2)c_a^2 = (0.887^2)(6.437) + (1-0.887^2)(1.0) = 5.27$$
- c_e^2 at machine 1 is large.

Step 5: **Why** is c_e^2 at machine 1 large?

- Effective CV at machine 1 is affected by failures,
  $$c_e^2 = c_0^2 + 2A(1-A)\frac{m_r}{t_0} = 0.25 + 6.18 = 6.43$$
- The inflation due to failures is large.
- Reducing MTTR at machine 1 would substantially improve performance.

Procoat Case – Situation

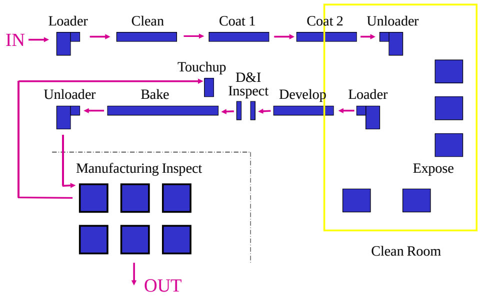
Problem:

- Current WIP around 1500 panels
- Desired capacity of 3000 panels/day (19.5 hr day with breaks/lunches)
- Typical output of 1150 panels/day
- Outside vendor being used to make up slack

Proposal:

- Expose is bottleneck, but in clean room
- Expansion would be expensive
- Suggested alternative is to add bake oven for touchups

Procoat Case – Layout



Procoat Case – Capacity Calculations

Machine Name	Process or Load Time (min)	Std Dev Process Time (min)	Conveyor Trip Time (min)	Number of Machines	MTTF	MTTR	Avail	Setup Time	Rate (p/day)	Time (min)
Clean1	0.33	0	15	1	80	4	0.95	0	3377	36.5
Coat1	0.33	0	15	1	80	4	0.95	0	3377	36.5
Coat2	0.33	0	15	1	80	4	0.95	0	3377	36.5
Expose	103	67	-	5	300	10	0.97	15	2879	121.9
Develop	0.33	0	2.67	1	300	3	0.99	0	3510	22.7
Inspect	0.5	0.5	-	2	-	-	1.00	0	4680	0.5
Bake	0.33	0	100	1	300	3	0.99	0	3510	121.0
MI	161	64	-	8	-	-	1.00	0	3488	161.0
Touchup	9	0	-	1	-	-	1.00	0	7800	9.0
									2879	545.7

r_b = 2,879 p/day
T_0 = 546 min = 0.47 days
$W_0 = r_b T_0$ = 1,343 panels

Procoat Case – Benchmarking

TH Resulting from PWC with WIP = 1,500:

$$TH = \frac{w}{w + W_0 - 1} r_b = \frac{1{,}500}{1{,}500 + 1{,}343 - 1} 2{,}879 = 1{,}520$$

Higher than actual TH

***Conclusion*: actual system is significantly worse than PWC.**

***Question*: what to do?**

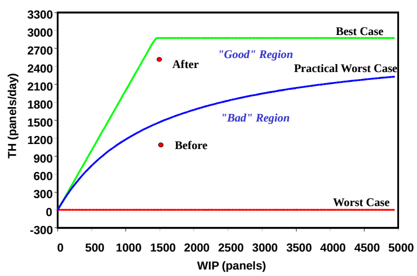
Procoat Case – Factory Physics® Analysis

1) Bottleneck Capacity (Expose)
- **rate: operator training, setup reduction**
- **time: break spelling, shift changes**

2) Bottleneck Starving
- **process variability: operator training**
- **flow variability: coater line – field ready replacements**

reduces "desire to queue" so that clean room buffer is adequate

Procoat Case – Outcome



Corrupting Influence Takeaways

Variance Degrades Performance:

- many sources of variability
- planned and unplanned

Variability *Must* be Buffered:

- inventory
- capacity
- time

Flexibility Reduces Need for Buffering:

- still need buffers, but smaller ones

Corrupting Influence Takeaways (cont.)

Variability and Utilization Interact:

- congestion effects multiply
- utilization effects are highly nonlinear
- importance of bottleneck management

Batching is an Important Source of Variability:

- process and move batching
- serial and parallel batching
- wait-to-batch time in addition to variability effects

Corrupting Influence Takeaways (cont.)

Assembly Operations Magnify Impact of Variability:

- wait-to-match time
- caused by lack of synchronization

Variability Propagates:

- flow variability is as disruptive as process variability
- non-bottlenecks can be major problems

End of Session

Thank You

Course is brought to you by

South Dakota School of Mines & Technology

In co-operation with

Electronic University Consortium of South Dakota