Chapter 9: The Corrupting Influence of Variability (continued)

Chapter 10: Push & Pull Production Systems
Agenda

Web Resources
Schedule
Factory Physics

(New Assignment
Chapter 8: Problem 6, 8
Chapter 9: Problems 1-4
Chapter 10: Problems 1, 2, 3, 5)
Web Resources

http://sdmines.sdsmt.edu/sdsmt/directory/courses/2009fa/tm663M021-099

I should have e-mailed the solutions through today’s assignment today.
<table>
<thead>
<tr>
<th>Date</th>
<th>Chapters</th>
<th>Assigned</th>
<th>Date</th>
<th>Chapters</th>
<th>Assigned</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/31/2009</td>
<td></td>
<td>0,1</td>
<td>11/23/2009</td>
<td></td>
<td>Exam 2</td>
</tr>
<tr>
<td>9/21/2009</td>
<td>2, 3</td>
<td>C3: 2,3,5,6,11</td>
<td>12/14/2009</td>
<td>Final</td>
<td></td>
</tr>
<tr>
<td>9/28/2009</td>
<td>4, 5</td>
<td>Study Q’s</td>
<td>12/14/2009</td>
<td>Final</td>
<td></td>
</tr>
<tr>
<td>10/5/2009</td>
<td>6, 7</td>
<td>C6:1 C7:5,8,11</td>
<td>18, 19 Not covered</td>
<td>We may rearrange a bit. We could skip chapter 11 &amp; 12 and do 18 &amp; 19</td>
<td></td>
</tr>
<tr>
<td>10/12/2009</td>
<td>Holiday</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10/19/2009</td>
<td>Exam 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10/26/2009</td>
<td>8, 9</td>
<td>C8:6,8 C9:1-4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11/2/2009</td>
<td>9, 10</td>
<td>C10:1,2,3,5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11/9/2009</td>
<td>11, 12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11/16/2009</td>
<td>13, 14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
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:
- $k = \text{parallel batch size (10)}$
- $t = \text{time to process a batch (90)}$
- $c_e = \text{CV for batch (1.0)}$
- $r_a = \text{arrival rate for parts (0.05)}$
- $c_a = \text{CV of batch arrivals (1.0)}$
- $B = \text{maximum batch size (100)}$

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

$W = \frac{10-1}{2} \frac{1}{0.05} = 90$

**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:

\[ WT = \frac{k-1}{2} \frac{1}{r_a} = \frac{10-1}{2} \frac{1}{0.05} = 90 \]

Average queue plus process time at station:

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

Total cycle time:

\[ CT + WT = 90 + 130.5 = 220.5 \]

Batch size affects both wait-for-batch time and queue time.
Cycle Time vs. Batch Size in a Parallel Operation

![Graph showing cycle time vs. batch size with annotations](image)

**queue time due to utilization**

**wait for batch time**

**Optimum Batch Size**

**Total Cycle Time**
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
- $CT = \frac{WIP}{TH} = \frac{150}{100} = \frac{3}{2} \text{ hr} = 90 \text{ min}$

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

- $CT = 0.5 \text{ min} = 0.0083 \text{ hr}$
- $WIP = TH \times CT = 100 \times 0.0083 = 0.83 \text{ wrenches}$

**Conclusion:** Dramatic reduction in WIP and CT due to small batches—
independent of variability or other factors.
Move Batching Law: 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 $k$?

\[ r_a, c_a(1) \quad t_e(1), c_e(1) \quad k \quad t_e(2), c_e(2) \]

Station 1

Station 2

single job

batch
Move Batching Calculations

Time at First Station:

- Average time before batching is:
  \[
  \frac{c_a^2(1) + c_e^2(1)}{2} \frac{u(1)}{1-u(1)} t_e(1) + t_e(1)\]

- Average time forming the batch is:
  \[
  \frac{k-1}{2} \frac{1}{r_a} = \frac{k-1}{2u(1)} t_e(1)\]

- Average time spent at the first station is:
  \[
  CT(1) = \frac{c_a^2(1) + c_e^2(1)}{2} \frac{u(1)}{1-u(1)} t_e(1) + t_e(1) + \frac{k-1}{2u(1)} t_e(1)\]
  \[
  = CT(1, \text{no batching}) + \frac{k-1}{2u(1)} t_e(1)\]
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
  
  $c_d^2(1) = (1 - u(1)^2)c_a^2(1) + u(1)^2 c_e^2(1)$

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

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

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

because

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

because departures are independent, so variances add

variance divided by mean squared...
Move Batching Calculations (cont.)

Time at Second Station:

- 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:

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

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

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

$$= CT(2, \text{no batching}) + \frac{k-1}{2}t_e(2)$$
Move Batching Calculations (cont.)

Total Cycle Time:

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

\[
= CT(\text{no batching}) + \left(\frac{k-1}{2}\right)\left(\frac{t_e(1)}{u(1)} + t_e(2)\right)
\]

Insight:

- Cycle time increases with \(k\).
- Inflation term does not involve CV’s.
- Congestion from batching is more bad control than randomness.

\(\text{inflation factor due to move batching}\)
**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 a special instance of the 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:*

\[
\text{cycle time} = \text{move time} + \text{queue time} + \text{setup time} + \\
\quad \text{process time} + \text{wait-to-batch time} + \\
\quad \text{wait-in-batch time} + \text{wait-to-match time}
\]

- 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

\[ CT_q = V \times U \times t \]

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

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

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

\[ CT_{batch} = \text{delay at stations} + \text{delay between stations} \]

**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

\[ CT_{\text{batch}} = \text{delay due to lack of synchronization} \]

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”

\[ CT_q = V \times U \times t \]

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

**Reduce Variability**
**Reduce Utilization**

*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

CT = 10
$\sigma_{CT} = 3$

CT = 10
$\sigma_{CT} = 6$

Lead Time = 14 days

Lead Time = 27 days
Diagnostics Using Factory Physics®

Situation:

- Two machines in series; machine 2 is bottleneck
- $c_a^2 = 1$
- Machine 1: $t_0 = 19$ min
  $c_0^2 = 0.25$
  MTTF = 48 hr, MTTR = 8 hr
- Machine 2: $t_0 = 22$ min
  $c_0^2 = 1$
  MTTF = 3.3 hr, MTTR = 10 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:**
\[
CT_q \approx \frac{c_a^2 + c_e^2}{2} \frac{u}{1-u} t_e
\]
\[
c_d^2 = u^2 c_e^2 + (1-u^2)c_a^2
\]

**Analysis:**

**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
  
  $$CT_q \approx \left( \frac{c_a^2 + c_e^2}{2} \right) \left( \frac{u}{1-u} \right) t_e = (3.16)(12.22)(23.11 \text{min})$$

  - 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,
  \[ c_e^2 = c_0^2 + 2 A(1 - A) \frac{m_r}{t_0} = 0.25 + 6.18 = 6.43 \]
- 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

IN → Loader → Clean → Coat 1 → Coat 2 → Unloader

Unloader → Bake → D&I Inspect → Develop → Loader

Manufacturing Inspect → Touchup

OUT

Clean Room → Expose → Loader

Frank Matejcik  SD School of Mines & Technology  33
Procoat Case – Capacity Calculations

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

\[ r_b = 2,879 \text{ p/day} \]
\[ T_0 = 546 \text{ min} = 0.47 \text{ days} \]
\[ W_0 = r_b T_0 = 1,343 \text{ panels} \]
Procoat Case – Benchmarking

TH Resulting from PWC with WIP = 1,500:

\[
TH = \frac{w}{w + W_0 - 1} r_b = \frac{1,500}{1,500 + 1,343 - 1} 2,879 = 1,520 \text{ Higher than actual TH}
\]

**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

![Graph showing Procoat Case Outcome]

- **Best Case**
- **Practical Worst Case**
- **Worse Case**
- **Before**
- **After**

- **"Good" Region**
- **"Bad" Region**

**TH (panels/day)** vs **WIP (panels)**

Frank Matejcik  SD School of Mines & Technology  37
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
Push and Pull Production Systems

You say yes.
I say no.
You say stop.
and I say go, go, go!

– The Beatles
The Key Difference Between Push and Pull

**Push Systems:** *schedule* work releases based on demand.

- inherently due-date driven
- control release rate, observe WIP level

**Pull Systems:** *authorize* work releases based on system status.

- inherently rate driven
- control WIP level, observe throughput
Push vs. Pull Mechanics

**PUSH**

(Exogenous) Schedule

Job

Production Process

Push systems do not limit WIP in the system.

**PULL**

(Endogenous) Stock Void

Job

Production Process

Pull systems deliberately establish a limit on WIP.
What Pull is Not!

Make-to-Order:
• MRP with firm orders on MPS is make-to-order.
• But it does not limit WIP and is therefore a push system.

Make-to-Stock:
• Pull systems do replenish inventory voids.
• But jobs can be associated with customer orders.

Forecast Free:
• Toyota’s classic system made cars to forecasts.
• Use of takt times or production smoothing often involves production without firm orders (and hence forecasts).
Push and Pull Examples

Are the following systems essentially push or essentially pull?

- Kinko’s copy shop: **PUSH**
- Soda vending machine: **PULL**
- “Pure” MRP system: **PUSH**
- Doctor’s office: **PUSH – into office, PULL into exam room**
- Supermarket (goods on shelves): **PULL**
- Tandem line with finite interstation buffers: **PULL**
- Runway at O’Hare during peak periods: **PULL**
- Order entry server at Amazon.com: **PUSH**
Push and Pull Line Schematics

Pure Push (MRP)

Pure Pull (Kanban)

CONWIP

Authorization Signals

Full Containers
Pulling with Kanban

Outbound stockpoint

Completed parts with cards enter outbound stockpoint.

Production cards

When stock is removed, place production card in hold box.

Outbound stockpoint

Production card authorizes start of work.
Inventory/Order Interface

Concept:

• Make-to-stock and make-to-order can be used in same system.
• Dividing point is called the inventory/order interface.
• This is sometimes called the push/pull interface, but since WIP could be limited or unlimited in both segments, this is not a strictly accurate term.

Benefit: eliminate entire portion of cycle time seen by customers by building to stock.

Implementation:

• kanban
• late customization (postponement)
Example – Custom Taco Production Line

I/O Interface

Make-to-Stock

Refrigerator

Make-to-Order

Cooking → Assembly → Packaging → Sales

Customer
Example – Quick Taco Production Line

Make-to-Order

I/O Interface

Make-to-Stock

Notes:
• I/O interface can differ by time of day (or season).
• I/O interface can differ by product.
Example – IBM Panel Plant

Original Line

Treater → [Prepreg, Copper] → Lamination → Machining → Circuitize → Drilling → Copper Plate → Procoat → Sizing, Test

I/O Interface

Revised Line

Treater → [Prepreg, Copper] → Lamination → Machining → [Core Blanks] → Circuitize → Drilling → Copper Plate → Procoat

I/O Interface

Notes:
- Moving I/O interface closer to customer shortens leadtime seen by customer.
- Small number of core blanks presents opportunity to make them to stock.
Example – HP Deskjet Supply Chain

Notes:
- I/O interface located in markets to achieve quick response to customers
- Delayed differentiation of products (power supplies for different countries) enables pooling of safety stocks
I/O Interface Conclusions

Basic Tradeoff:

- responsiveness vs. inventory (time vs. money)
- moving PPI closer to customer increases responsiveness and (usually) inventory

Optimal Position of I/O Interface:

- need for responsiveness
- cost of carrying inventory \(\Rightarrow\) product diversification

Levers:

- product design (postponement)
- process design (quick response manufacturing)
Advantages of Pull Systems

**Low Unit Cost:**
- low inventory
- reduced space
- little rework

**High External Quality:**
- high internal quality
- pressure for good quality
- promotion of good quality (e.g., defect detection)

**Good Customer Service:**
- short cycle times
- steady, predictable output stream

**Flexibility:**
- avoids committing jobs too early
- encourages floating capacity
The Magic of Pull

Pulling Everywhere?

You don’t never make nothin’ and send it no place. Somebody has to come get it.

– Hall 1983

No! It’s the WIP Cap:

• Kanban – WIP cannot exceed number of cards
• “WIP explosions” are impossible
Pull Benefits Achieved by WIP Cap

Reduces Costs:
- prevents WIP explosions
- reduces average WIP
- reduces engineering changes

Improves Quality:
- pressure for higher quality
- improved defect detection
- improved communication

Improves Customer Service:
- reduces cycle time variability
- pressure to reduce sources of process variability
- promotes shorter lead times and better on-time performance

Maintains Flexibility:
- avoids early release (like air traffic control)
- less direct congestion
- less reliance on forecasts
- promotes floating capacity
CONWIP

Assumptions:
1. Single routing
2. WIP measured in units

Mechanics: allow next job to enter line each time a job leaves (i.e., maintain a WIP level of $m$ jobs in the line at all times).

Modeling:
- MRP looks like an open queueing network
- CONWIP looks like a closed queueing network
- Kanban looks like a closed queueing network with blocking
CONWIP vs. Pure Push

Push/Pull Laws: A CONWIP system has the following advantages over an equivalent pure push system:

1) **Observability**: WIP is observable; capacity is not.

2) **Efficiency**: A CONWIP system requires less WIP on average to attain a given level of throughput.

3) **Robustness**: A profit function of the form

   \[ \text{Profit} = p \text{Th} - h \text{WIP} \]

   is more sensitive to errors in TH than WIP.
CONWIP Efficiency Example

Equipment Data:
- 5 machines in tandem, all with capacity of one part/hr \((u=TH \cdot t_e = TH)\)
- exponential (moderate variability) process times

CONWIP System:

\[
TH(w) = \frac{w}{w + W_0 - 1} \quad r_b = \frac{w}{w + 4}
\]

Pure Push System:

\[
w(TH) = 5 \frac{u}{1 - u} = 5 \frac{TH}{1 - TH}
\]
CONWIP Efficiency Example (cont.)

How much WIP is required for push to match TH attained by CONWIP system with WIP=$w$?

\[
w \left( \frac{w}{w+4} \right) = \frac{5(w/(w+4))}{1 - (w/(w+4))} = \frac{5w}{4}
\]

• *In this example, WIP is always 25% higher for same TH in push than in CONWIP*

• *In general, the increase won’t always be 25%, but it will always take more WIP to get same TH under push than under pull.*
CONWIP Robustness Example

Profit Function: Profit = \( pTH - hw \)

CONWIP: Profit(\( w \)) = \( p \left( \frac{w}{w + 4} \right) - hw \)  
need to find “optimal”  
WIP level

Push: Profit(\( TH \)) = \( pTH - h \left( \frac{5TH}{1-TH} \right) \)  
need to find “optimal”  
TH level (i.e., release rate)

Key Question: what happens when we don’t choose optimum  
values (as we never will)?
CONWIP vs. Pure Push Comparisons

Control as Percent of Optimal

Efficiency

Robustness
Modeling CONWIP with Mean-Value Analysis

Notation:

\[ u_j(w) = \text{utilization of station } j \text{ in CONWIP line with WIP level } w \]
\[ CT_j(w) = \text{cycle time at station } j \text{ in CONWIP line with WIP level } w \]
\[ CT(w) = \sum_{j=1}^{n} CT_j(w) = \text{cycle time of CONWIP line with WIP level } w \]
\[ TH(w) = \text{throughput of CONWIP line with WIP level } w \]
\[ WIP_j(w) = \text{average WIP level at station } j \text{ in CONWIP line with WIP level } w \]

**Basic Approach:** Compute performance measures for increasing \( w \) assuming job arriving to line “sees” other jobs distributed according to average behavior with \( w-1 \) jobs.
Mean-Value Analysis Formulas

Starting with $WIP_j(0)=0$ and $TH(0)=0$, compute for $w=1,2,…$

$$CT_j(w) = \frac{t_e^2(j)}{2}[c_e^2(j) - 1]TH(w-1) + [WIP_j(w-1) + 1]t_e(j)$$

$$CT(w) = \sum_{j=1}^{n} CT_j(w)$$

$$TH(w) = \frac{w}{CT(w)}$$

$$WIP_j(w) = TH(w)CT_j(w)$$
## Computing Inputs for MVA

<table>
<thead>
<tr>
<th>MEASURE:</th>
<th>STATION:</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Natural Process Time (hr)</td>
<td>( t_0 )</td>
<td>0.090</td>
<td>0.090</td>
<td>0.098</td>
<td>0.090</td>
<td>0.090</td>
</tr>
<tr>
<td>Natural Process CV</td>
<td>( c_0^2 )</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Number of Machines</td>
<td>( m )</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>MTTF (hr)</td>
<td>( m_r )</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>MTTR (hr)</td>
<td>( m_r )</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Availability</td>
<td>( A )</td>
<td>0.990</td>
<td>0.995</td>
<td>0.990</td>
<td>0.990</td>
<td>0.990</td>
</tr>
<tr>
<td>Effective Process Time (failures only)</td>
<td>( t_e' )</td>
<td>0.091</td>
<td>0.090</td>
<td>0.099</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>Eff Process CV (failures only)</td>
<td>( c_e^2' )</td>
<td>0.436</td>
<td>0.110</td>
<td>0.400</td>
<td>0.436</td>
<td>0.436</td>
</tr>
<tr>
<td>Jobs Between Setups</td>
<td>( N_s )</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
<td>100.000</td>
</tr>
<tr>
<td>Setup Time (hr)</td>
<td>( t_s )</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>Setup Time CV</td>
<td>( c_s^2 )</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>0.500</td>
</tr>
<tr>
<td>Eff Process Time (failures+setups)</td>
<td>( t_e )</td>
<td>0.091</td>
<td>0.090</td>
<td>0.099</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>Eff Station Rate</td>
<td>( r_e )</td>
<td>11.001</td>
<td>11.056</td>
<td>10.103</td>
<td>11.001</td>
<td>11.001</td>
</tr>
<tr>
<td>Eff Process Time Var (failures+setups)</td>
<td>( \sigma_e^2 )</td>
<td>0.004</td>
<td>0.001</td>
<td>0.004</td>
<td>0.004</td>
<td>0.004</td>
</tr>
<tr>
<td>Eff Process CV (failures+setups)</td>
<td>( c_e^2 )</td>
<td>0.436</td>
<td>0.110</td>
<td>0.400</td>
<td>0.436</td>
<td>0.436</td>
</tr>
<tr>
<td>( r_b )</td>
<td>10.103</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( T_0 )</td>
<td>0.462</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( W_0 )</td>
<td>4.669</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TH</td>
<td>CT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Output of MVA

<table>
<thead>
<tr>
<th>w</th>
<th>TH</th>
<th>CT</th>
<th>CT&lt;sub&gt;1&lt;/sub&gt;(w)</th>
<th>CT&lt;sub&gt;2&lt;/sub&gt;(w)</th>
<th>CT&lt;sub&gt;3&lt;/sub&gt;(w)</th>
<th>CT&lt;sub&gt;4&lt;/sub&gt;(w)</th>
<th>CT&lt;sub&gt;5&lt;/sub&gt;(w)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Actual</td>
<td>Actual</td>
<td>0.091</td>
<td>0.090</td>
<td>0.099</td>
<td>0.091</td>
<td>0.091</td>
</tr>
<tr>
<td>1</td>
<td>2.164</td>
<td>0.462</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3.807</td>
<td>0.525</td>
<td>0.104</td>
<td>0.100</td>
<td>0.114</td>
<td>0.104</td>
<td>0.104</td>
</tr>
<tr>
<td>3</td>
<td>5.037</td>
<td>0.596</td>
<td>0.118</td>
<td>0.111</td>
<td>0.131</td>
<td>0.118</td>
<td>0.118</td>
</tr>
<tr>
<td>4</td>
<td>5.957</td>
<td>0.672</td>
<td>0.133</td>
<td>0.123</td>
<td>0.149</td>
<td>0.133</td>
<td>0.133</td>
</tr>
<tr>
<td>5</td>
<td>6.651</td>
<td>0.752</td>
<td>0.149</td>
<td>0.135</td>
<td>0.170</td>
<td>0.149</td>
<td>0.149</td>
</tr>
<tr>
<td>6</td>
<td>7.185</td>
<td>0.835</td>
<td>0.166</td>
<td>0.147</td>
<td>0.191</td>
<td>0.166</td>
<td>0.166</td>
</tr>
<tr>
<td>7</td>
<td>7.604</td>
<td>0.921</td>
<td>0.182</td>
<td>0.160</td>
<td>0.214</td>
<td>0.182</td>
<td>0.182</td>
</tr>
<tr>
<td>8</td>
<td>7.938</td>
<td>1.008</td>
<td>0.199</td>
<td>0.173</td>
<td>0.237</td>
<td>0.199</td>
<td>0.199</td>
</tr>
<tr>
<td>9</td>
<td>8.210</td>
<td>1.096</td>
<td>0.216</td>
<td>0.186</td>
<td>0.262</td>
<td>0.216</td>
<td>0.216</td>
</tr>
<tr>
<td>10</td>
<td>8.435</td>
<td>1.186</td>
<td>0.233</td>
<td>0.198</td>
<td>0.288</td>
<td>0.233</td>
<td>0.233</td>
</tr>
<tr>
<td>11</td>
<td>8.624</td>
<td>1.276</td>
<td>0.250</td>
<td>0.211</td>
<td>0.315</td>
<td>0.250</td>
<td>0.250</td>
</tr>
<tr>
<td>12</td>
<td>8.784</td>
<td>1.366</td>
<td>0.267</td>
<td>0.224</td>
<td>0.342</td>
<td>0.267</td>
<td>0.267</td>
</tr>
<tr>
<td>13</td>
<td>8.922</td>
<td>1.457</td>
<td>0.283</td>
<td>0.236</td>
<td>0.371</td>
<td>0.283</td>
<td>0.283</td>
</tr>
<tr>
<td>14</td>
<td>9.041</td>
<td>1.549</td>
<td>0.300</td>
<td>0.249</td>
<td>0.400</td>
<td>0.300</td>
<td>0.300</td>
</tr>
<tr>
<td>15</td>
<td>9.145</td>
<td>1.640</td>
<td>0.316</td>
<td>0.261</td>
<td>0.430</td>
<td>0.316</td>
<td>0.316</td>
</tr>
</tbody>
</table>
Using MVA to Evaluate Line Performance
Implementing Pull

**Pull is Rigid:** replenishing stocks quickly (just in time) requires level mix, volume, sequence.

**JIT Practices:**

- **Support Rigidity:**
  - production smoothing/mix stabilization
- **Mitigate Rigidity in Production System**
  - capacity buffers
  - setup reduction
  - flexible labor
  - facility layout
  - product design (postponement, etc.)
- **Mitigate Rigidity in Organization**
  - TQM
  - vendor management, etc.

*This is the “genius” of pull!*
Capacity Buffers

Motivation: facilitate rapid replenishments with minimal WIP

Benefits:
• Protection against quota shortfalls
• Regular flow allows matching against customer demands
• Can be more economical in long run than WIP buffers in push systems

Techniques:
• Planned underutilization (e.g., use $u = 75\%$ in aggregate planning)
• *Two shifting*: 4 – 8 – 4 – 8
• Schedule dummy jobs to allow quick response to hot jobs
Setup Reduction

**Motivation:** Small lot sequences not feasible with large setups.

**Internal vs. External Setups:**
- External – performed while machine is still running
- Internal – performed while machine is down

**Approach:**
1. Separate the internal setup from the external setup.
2. Convert as much internal setup as possible to external setup.
3. Eliminate the adjustment process.
4. Abolish the setup itself (e.g., uniform product design, combined production, parallel machines).
Flexible Labor

Cross-Trained Workers:

• float where needed
• appreciate line-wide perspective
• provide more heads per problem area

Shared Tasks:

• can be done by adjacent stations
• reduces variability in tasks, and hence line stoppages/quality problems

work can float to workers, or workers can float to work…
Cellular Layout

**Advantages:**
- Better flow control
- Improved material handling (smaller transfer batches)
- Ease of communication (e.g., for floating labor)

**Challenges:**
- May require duplicate equipment
- Product to cell assignment

Inbound Stock  Outbound Stock
Focused Factories

Pareto Analysis:
- Small percentage of sku’s represent large percentage of volume
- Large percentage of sku’s represent little volume but much complexity

Dedicated Lines:
- for families of high runners
- few setups
- can use pull effectively

Job Shop Environment:
- for low runners
- many setups
- poorer performance, but only on smaller portion of business
- may need to use push
Push/Pull Takeaways

Magic of Pull: the WIP cap

MTS/MTO Hybrids: locating the I/O interface

Logistical Benefits of Pull:
  • observability
  • efficiency
  • robustness (this is the key one)

Overcoming Rigidity of Pull:
  • capacity buffers
  • setup reduction
  • flexible labor
  • facility layout, etc.