Performance analysis of manufacturing systems under hybrid control policies

Asbjoern M. Bonvik

Doctoral Thesis Proposal

Oct 3, 1995

Outline

- Objectives
- Previous work
  - Control policies
  - Performance analysis
- Motivational example
- Decompositions
  - Building blocks
  - Kanban
  - CONWIP/finite buffer hybrid policy
  - Restricted basestock/finite buffer hybrid policy
- Remaining work
- Timetable

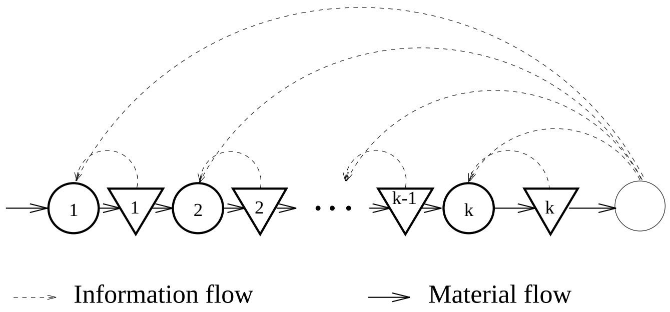
Objectives:

- To analyze the performance of production/inventory systems operated by a variety of control policies.
- To construct tools for fast estimation of main performance measures of such systems.
- This will eventually permit optimization over the space of possible control policies and the parameters of these policies.

Control policies

- Transfer line - finite buffers, unlimited demand.
- Kanban - finite buffers, synchronized to demand process.
  (There are several versions of kanban policies.)
- Basestock - work orders synchronize each stage to demand.
- CONWIP - single kanban cell containing entire line.
- Hedging point policies - Similar to basestock.
  (from optimal control formulations)
- Two-boundary control: hedging points and finite buffers.
- Other combined schemes: PAC, generalized kanban.

Unified view of control policies



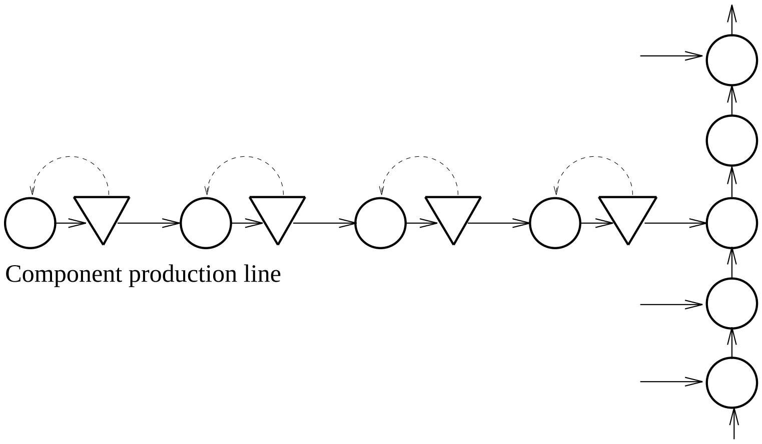
Two kinds of information flow:

- Local, between a buffer and an adjacent machine (kanban).
- Global, from demand process to every machine (basestock).
  Can be sent when a demand arrives or when it is satisfied.

Performance analysis methods

- Discrete event simulation
  Pro: Can in principle be made arbitrarily accurate.
  Con: Time consuming to develop and use.
- Queuing theoretic methods
  Pro: Gives deep understanding of system.
  Con: Only exact for a limited class of systems.
  Mathematics can be prohibitive.
- Decompositions
  Pro: Fast and simple.
  Con: Approximate results.

A component production line feeding final assembly



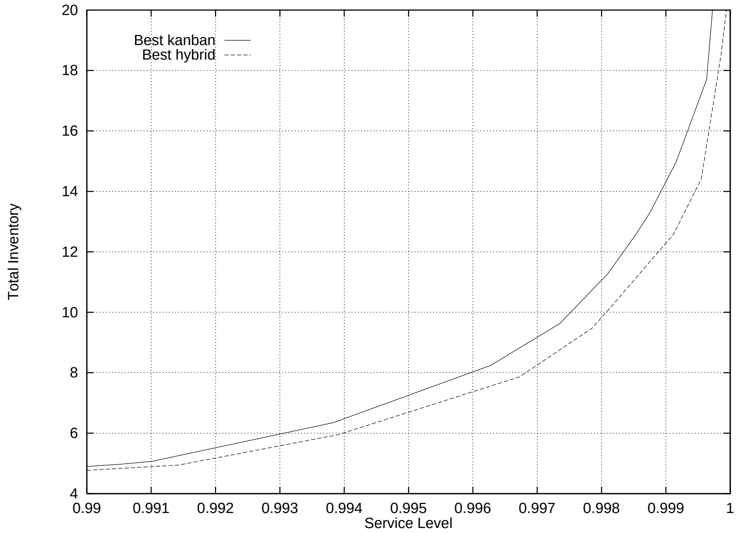
Example based on an actual line at the Toyota automobile assembly factory in Nagoya, Japan.

Best service - inventory tradeoffs

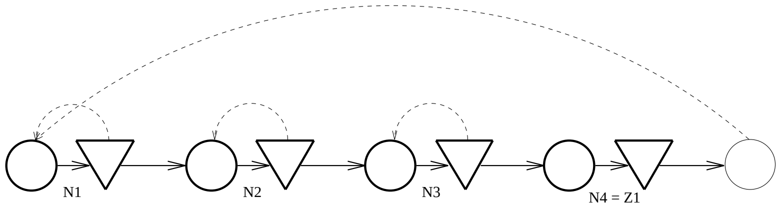
Objective: 99.9% service with minimal WIP

Best policy among:	WIP:
Kanban	15.82
Basestock	14.60
CONWIP	14.59
Hybrid	13.93

Throughput versus inventory



Recommended CONWIP - kanban hybrid



Only structural change:
Pass kanban from assembly line
to the *first* machine instead of the last.

Objectives:

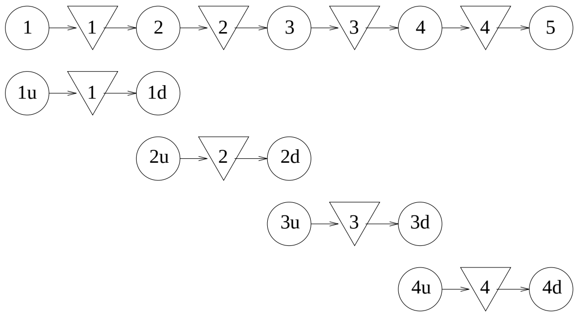
- To estimate:
  - throughput
  - average buffer levels
    (including finished goods and backlogged demand)
  - fraction of demand served from stock
  - average demand waiting time

- ... and to do so quickly.

Methodology:

- Find tractable building blocks, and solve these.
- Derive decomposition equations relating the building blocks.
- Construct efficient algorithms for solving the equations.
- Validate by comparing to simulation results.

Decomposition principles



Entire line intractable for analysis.

Can analyze two-machine lines.

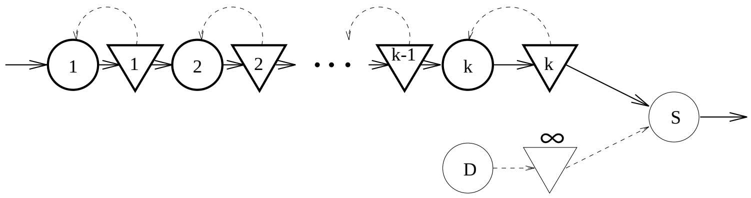
Focus on observed buffer behavior.

Describe line up- and downstream of a buffer by a single machine.

The parameters of these ‘virtual machines’ are unknown.

There is an efficient algorithm for finding these unknowns.

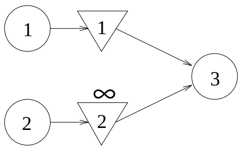
Finite buffer kanban model



Demand structure modeled as a machine.

Fast and reliable synchronization ‘machine.’

Solving the synchronization stage

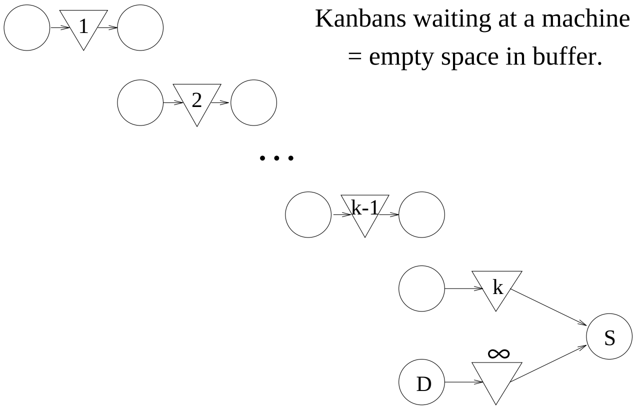


Machine 1 unreliable, can be up or down.
Synchronization ‘machine’ 3 reliable.
Demand ‘machine’ 2 unreliable.
Machines 1 and 2 are never starved.
Machine 3 is never blocked.

A continuous time, mixed state Markov process.
4 discrete states: Repair state of the two machines.
1 continuous state variable:
The level of buffer 1 - the level of buffer 2.
(buffer levels cannot be positive simultaneously)

The mathematics are the same as for a tandem 2-machine line!!!
(but we need to calculate a few extra integrals when it is solved.)

Finite buffer kanban model decomposed



Kanbans waiting at a machine
= empty space in buffer.

Can be solved by the same algorithm as a tandem line,
just using a single building block that is different.

Decomposition equations:

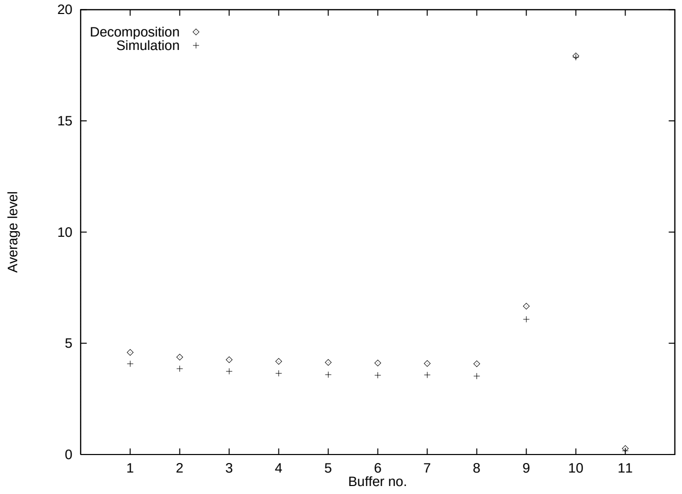
Unknowns: The failure and repair rates of the machines.
There are k buffers and 2 machines per buffer.

Equations:

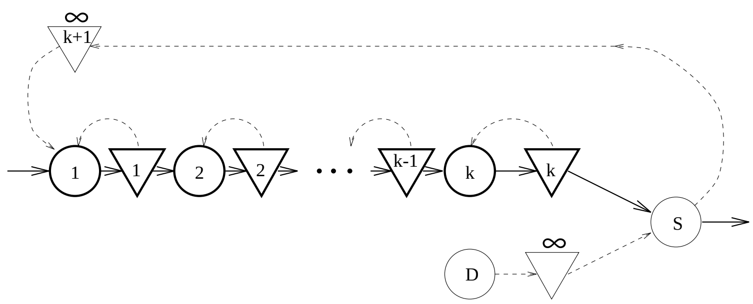
- Resumption of flow, matching the repair rates of 'machines' in the decomposition to the mean time to resume flow after an interruption in the line.
- Conservation of flow, ensuring that all stages have the same average throughput.
- Boundary conditions.

Gives 4 (k - 1) equations in 4 (k - 1) unknowns.

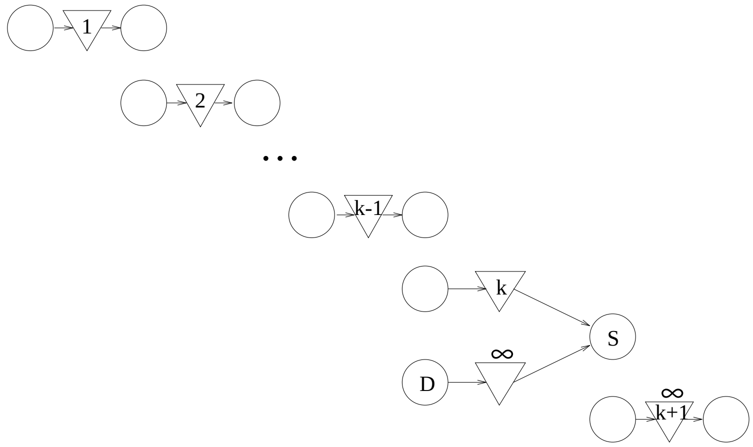
Buffer level accuracy, kanban decomposition



CONWIP/finite buffer hybrid system



CONWIP/finite buffer hybrid system decomposed



Decomposition equations:

Unknowns: The failure and repair rates of the machines.
There are k+1 buffers and 2 machines per buffer.
Equations: As before.
Gives 4 k - 2 equations in 4 k - 2 unknowns.

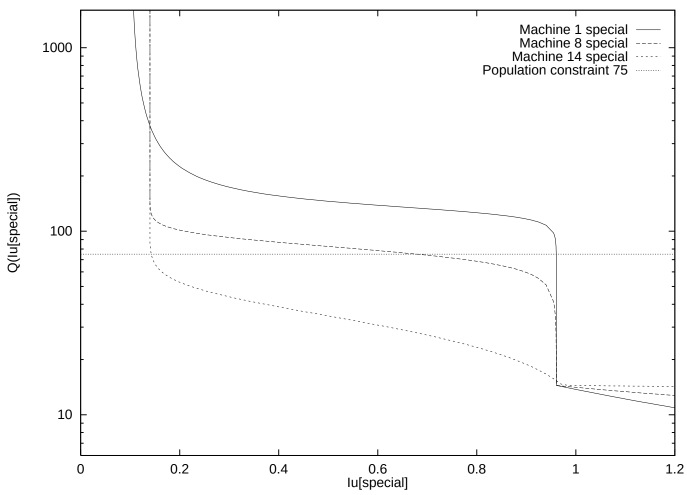
...but there are only 4 k - 3 independent equations.

The population constraint gives another equation.
The resulting system of equations is hard to solve.
We relax the population constraint to require that only
the sum of the average buffer levels matches the constraint.

Solution algorithm

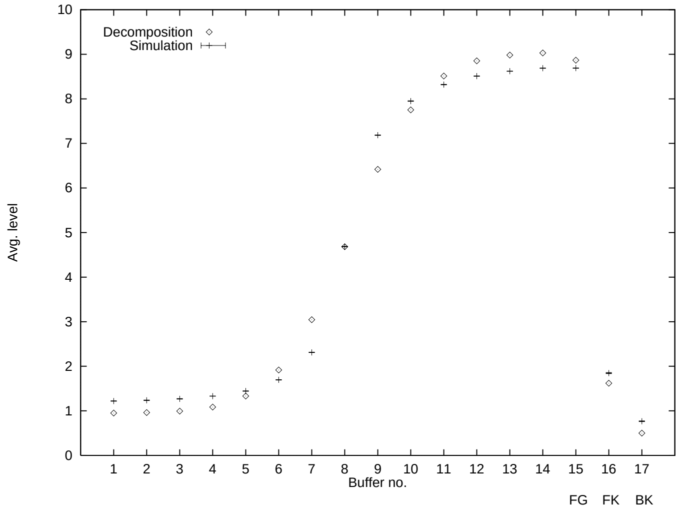
1. Select an unknown and fix its value.
   We used the repair rate of the upstream machine in some two-machine line.
2. Ignore the population constraint and perform decomposition as in the kanban case.
3. Sum average buffer levels and compare to constraint.
4. Adjust the unknown from step 1 accordingly.
5. Repeat steps 2 - 4 until the sum is sufficiently close to the population constraint.

Choice of ‘special’ machine is important!



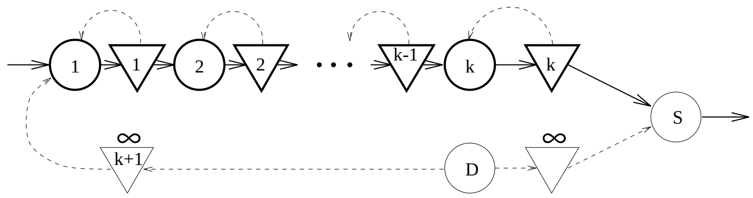
A.M.Bonvik — Thesis Proposal Meeting — Oct 3, 1995

The accuracy is pretty good for long lines



A.M.Bonvik — Thesis Proposal Meeting — Oct 3, 1995

Basestock/finite buffer hybrid



First machine is notified when demand arrives.

Restriction: All base stock levels zero, except for finished goods.

Work to be done:

- Extending existing work to inhomogeneous systems.
- Solving the restricted basestock/finite buffer model.
- Solving a detailed Markov process description
  of a small system to better understand sources of errors.
  Results inaccurate if few machines or low population constraint.
  Correlation effects?

Inhomogeneous systems

Already done:

- Decomposition of inhomogeneous closed loop systems.
- Continuous material simulation model for comparisons.

Preliminary results are encouraging.

Remaining:

- Synchronization stage solution and decomposition equations.
- Implementation and experimentation.

Small systems

Already done:

- Solved a discrete state Markov process representation of a two machine line with exponential processing times and Poisson demand.

Results are very close to simulation.

- Derived partial differential equations for continuous material case.

Remaining:

- Solving PDE's.

Proposed thesis outline

1. Introduction
2. Previous work
2.1 Control policies
2.2 Performance analysis
3. Motivational example
4 Decompositions
4.1 Building blocks
4.2 Kanban
4.3 CONWIP/finite buffer hybrid policy
4.4 Restricted basestock/finite buffer hybrid policy
5. A two-machine line in detail
6. Conclusions

Timetable

Oct 15, 1995 Inhomogeneous decomposition implemented.

Nov 1, 1995 Basestock/finite buffer decomposition done.

Dec 1, 1995 Numerical solver for small systems done.

Feb 1, 1996 Numerical results complete.

April, 1996 Thesis defense.

May 3, 1996 Thesis due date.