Synchronized Base Stock Policies

Ton de Kok
Department of Technology Management
Technische Universiteit Eindhoven

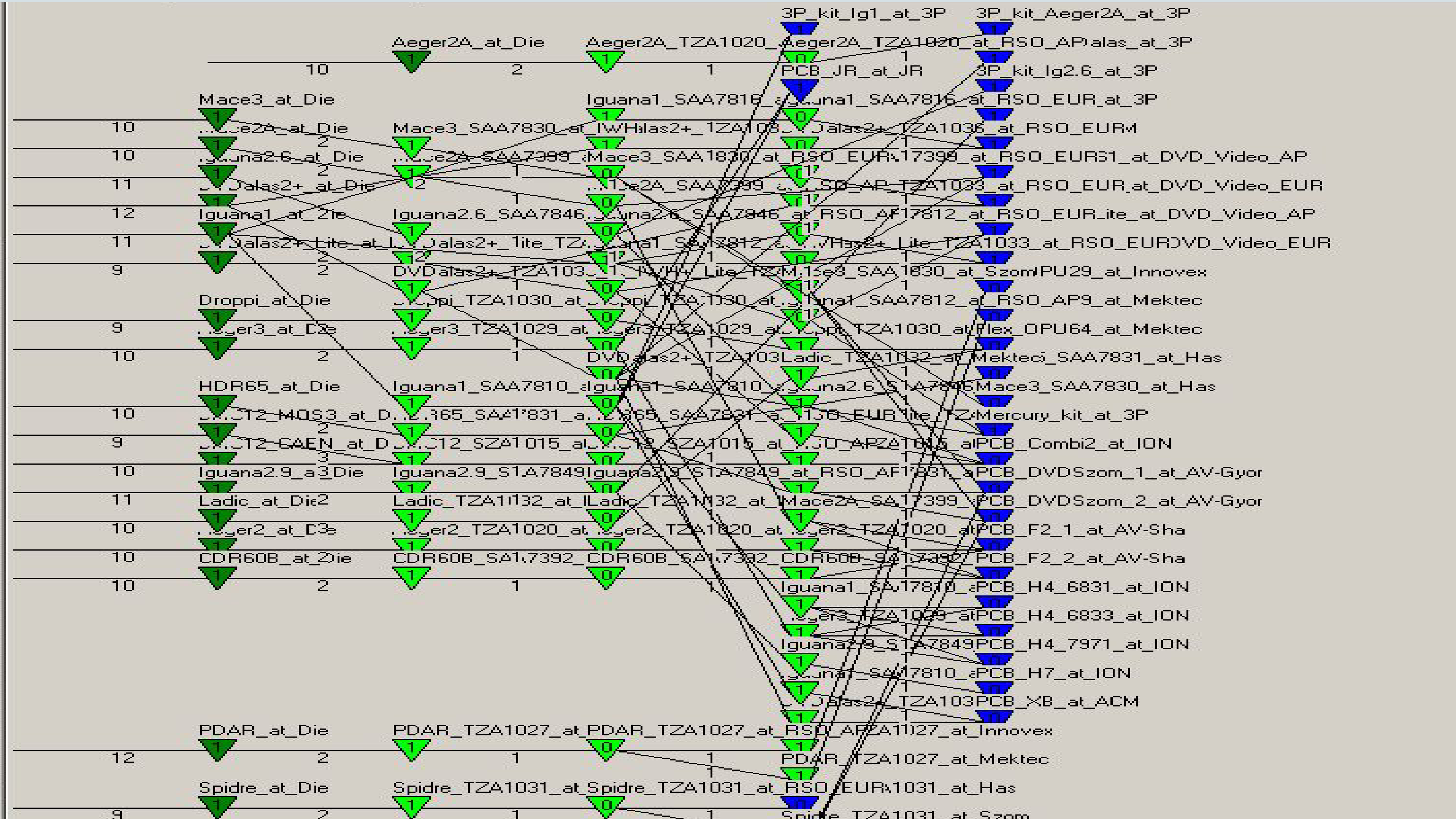
Contents

- Value Network Model
- Network transformation principles
- Divergent structures and allocation rules
- Optimality conditions on base stock levels
- Computational aspects
- Comparison against LP rolling schedule
- Managerial insights
- Conclusions and further research

Value Network Model

TU/e technische universiteit eindhoven

Beta Research School for Operations Management and Logistics



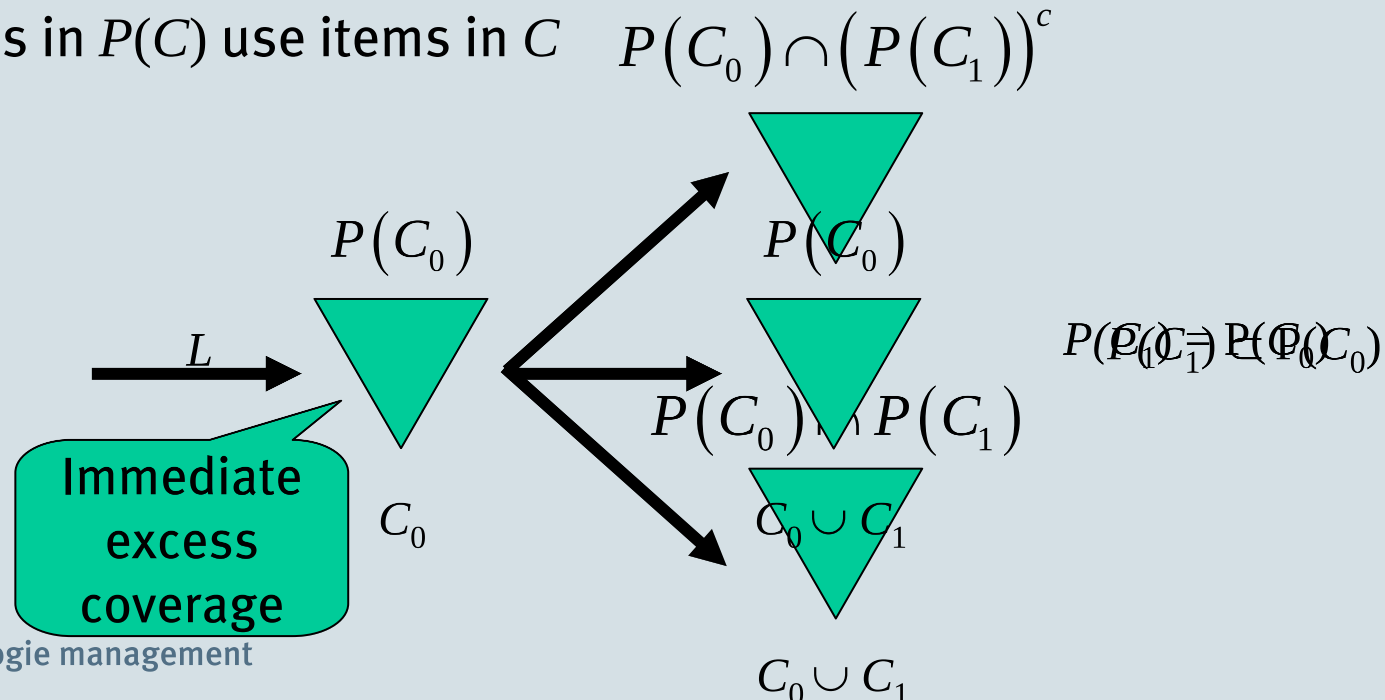
Multi-item multi-echelon inventory systems

- Arbitrary BOM
  - no alternative recipees, no multi-sourcing
- Stochastic end-item demand
  - I.i.d. periodic
- Linear holding costs
- Customer service objectives
  - End-item linear shortage costs
  - End-item service level constraints
- Item order lead times
  - Constant (planned)
- Unsatisfied end-item demand is backordered

Network transformation principles

Basic principle: Divergent node structures

- At time 0 item(s) with longest cumulative lead time has (have) been ordered
- At time t items $m \in C_0$ have been ordered, of which some at time t
- At time $t+L$ another set of items C_1 is ordered
- End-items in $P(C)$ use items in C



Product structure requirements for SBS-policies

- Acyclic BOM
- If two components i_1 and i_2 are common to items j_1 and j_2 then

$$\frac{a_{i_1 j_1}}{a_{i_2 j_1}} = \frac{a_{i_1 j_2}}{a_{i_2 j_2}} = c_{i_1 i_2}$$

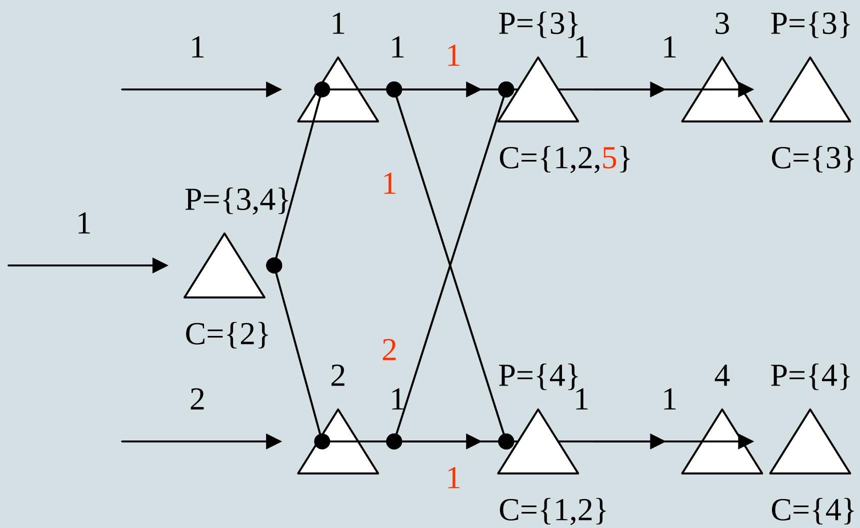
mutually non-zero elements of rows i_1 and i_2 are dependent

- Condition allows for elimination of i_2 with respect to (j_1, j_2)

$$\frac{a_{i_1 j_1} D_{j_1} + a_{i_1 j_2} D_{j_2}}{a_{i_2 j_1} D_{j_1} + a_{i_2 j_2} D_{j_2}} = c_{i_1 i_2}$$

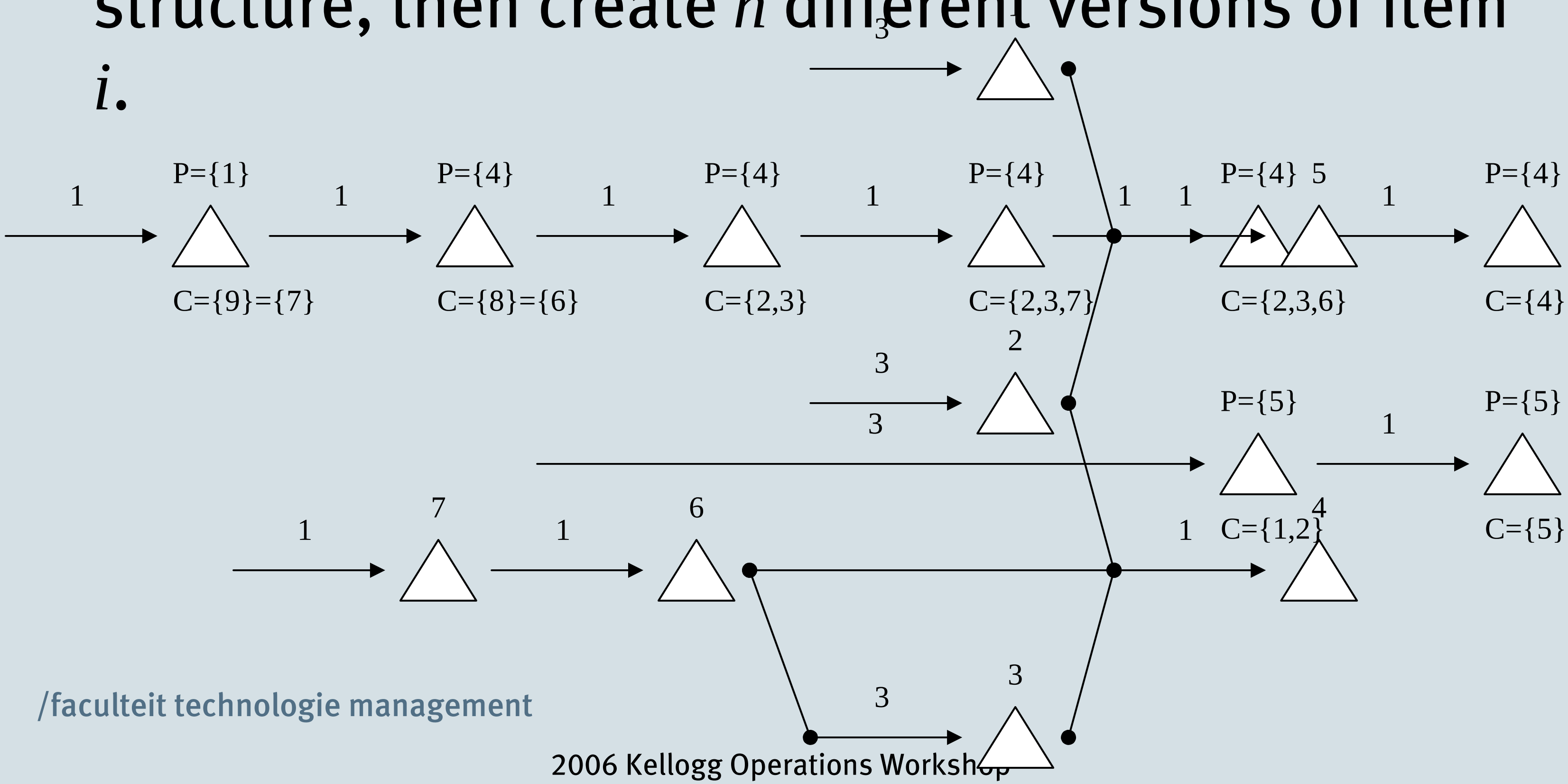
Elimination of different consumption ratios

- Create versions of items, whereby sum of consumption rates of versions satisfy the original product structure constraint



Elimination of cycles in BOM

- If item *i* is used at *n* different levels in the BOM structure, then create *n* different versions of item *i*.



Natural hierarchy

- BOM and lead time structures determine a natural hierarchy defined by divergent structurs
  – Divergent structures relate sets of end-items to sets of items
- Divergent structures determine forecasting requirements
  – An order of a set of items associated with a *node* can be released based on
    - the forecast of demand for the set of end-items associated with this node over its cumulative lead time (plus one period)
    - Safety stock requirements over the cumulative lead time

Dead stocks

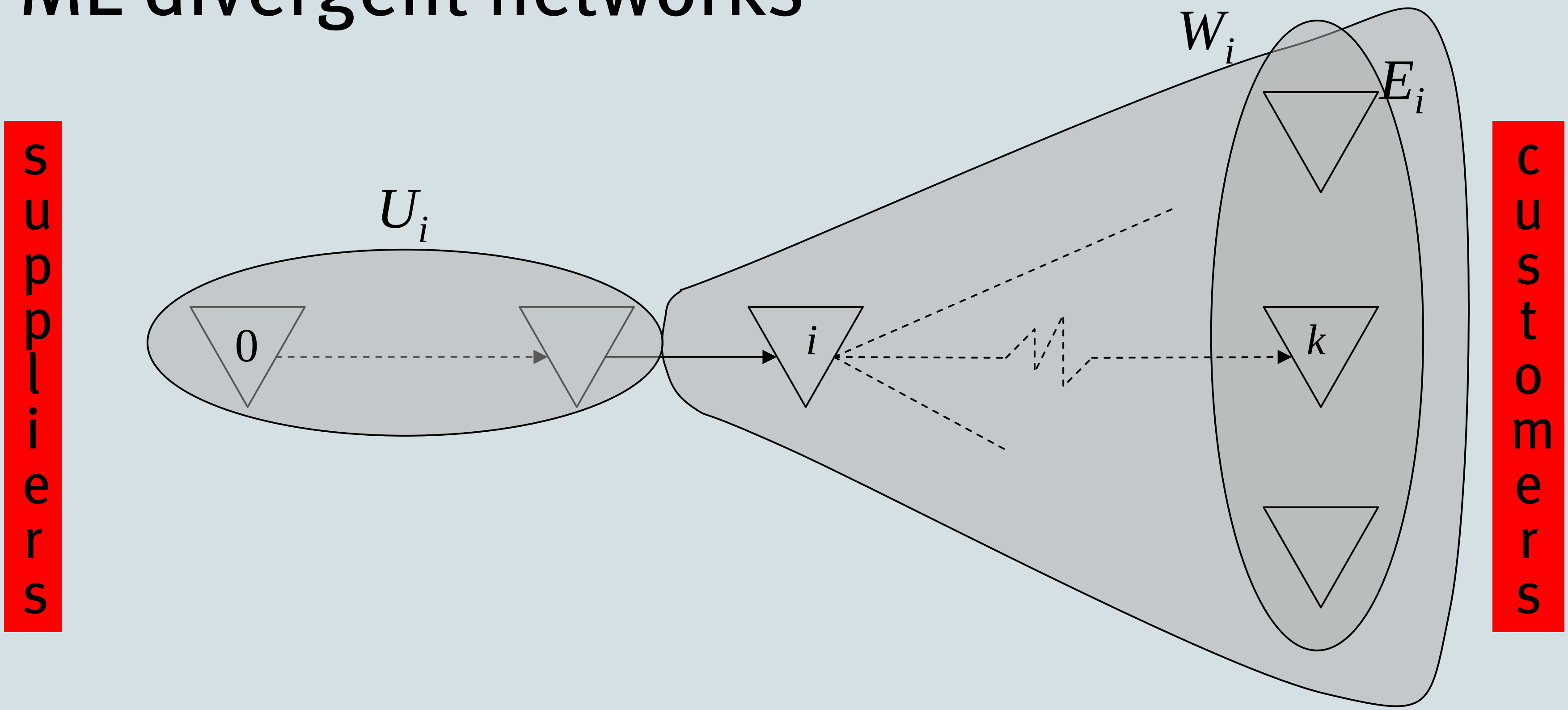
average stock in divergent structures

$$E\left[X_i\right] = \sum_{\{C \mid i \in C\}} E\left[\hat{X}_C^{P(C)}\right] + E\left[X_i^{dead}\right]$$

- Due to synchronization mechanism not all combinations of average stocks of items are possible
- If average inventories are taken from real life data and SBS policies are used, dead stock may result
  - net stock of an item cannot get below some positive number
- Computation of performance for given average stock levels requires determination of dead stocks
  - there is no unique solution to this
  - minimize value of dead stock

Divergent structures and allocation rules

ME divergent networks



Definitions

h_i	added value created (holding cost) at item i
p_k	penalty cost per unit short of end-item k
U_i	set of predecessors of i, i.e. all child items
W_i	set of all items in the echelon of i
E_i	set of all end-items in the echelon of i
Y_i	echelon inventory position of i
X_i	net stock of item i
I_{ki}	net stock of end-item k in a system with root node i

Base stock policies and linear allocation rules

Shortfall of i

$$Y_j(t) = S_j - q_j\left(Z_i(t-L_i) + D_i\left(t-L_i, t\right] - \Delta_j\right)^+$$

Parameter that determines stock kept at i on behalf of j

End-item demand during lead time of i

$$S_i = \sum_{j\in V_i}\left(S_j + q_j\Delta_j\right)$$

- j-specific Δ eliminates the effect of zero-cost successors on non-zero-cost successors: no stock of item i

$$E\left[X_i\right] = \sum_{j\in V_i} q_j E\left[\left(\Delta_j - Z_i - D_i\left(0, L_i\right]\right)^+\right]$$

Allocation fractions

- Eppen and Schrage (1981): equal stockout probability

$$q_j = \frac{\sigma_j}{\sum_{m \in V_i} \sigma_m}$$

- Van der Heijden et al (1997): minimal imbalance

$$q_j = \frac{\sigma_j^2}{2\sum_{j \in V_i} \sigma_m^2} + \frac{\mu_j^2}{2\sum_{j \in V_i} \mu_m^2}$$

Optimization problem

- Given allocation fractions, determine base stock levels for all items
- Expressions for costs and performance are similar to expressions for serial systems
- Imbalance should be sufficiently low to ensure validity of analytical results
  - in case imbalance too high, base stock levels should be such that average stocks are increased at upstream stages with high imbalance
  - requires experimental research

Optimality conditions on base stock levels

Optimal order-up-to-policies for divergent MEIS:

Generalized Newsvendor equations theorem for periodic review systems without setup costs

Optimal order-up-to-policies and allocation functions under balance assumption satisfy generalized Newsvendor equations:

Probability of non-stockout at downstream stockpoint k in the divergent subsystem with stockpoint i as most upstream stage equals

$$P\{I_{ki} \geq 0\} = \frac{p_k + \sum_{j \in U_i} h_j}{p_k + h_k + \sum_{j \in U_k} h_j}$$

or equivalently

$$\left(p_k + \sum_{j \in U_i} h_j \right) = \left(p_k + h_k + \sum_{j \in U_k} h_j \right) P\{I_{ki} \geq 0\}$$

Sample path condition on control policies

Assume that each node i is controlled according to a policy, which is defined by a parameter ξ_i and possibly other parameters and functions

Sample path condition

If ξ_j is increased by $|E_j|\varepsilon$ for all $j \in W_j$, and there is upstream availability then a one-time additional flow of ε units is created towards each $k \in E_j$

Main theorem

- If sample path condition holds then $\forall i \in I$

$$\sum_{k \in E_i} \left(p_k + \sum_{j \in U_k \setminus W_i} h_j \right) = \sum_{k \in E_i} \left(p_k + h_k + \sum_{j \in U_k} h_j \right) P\{ I_{ki} \geq 0 \}$$

- Proof based on echelon costs and similar to proof of single-item Newsvendor equation
- Relationship can be used as heuristic

Computational aspects

Finite horizon ruin probabilities

- Non-stockout probabilities $P\{I_{ki} \geq 0\}$ can be written as finite horizon ruin probabilities

$$P\{I_{ki} \geq 0\} = P\left\{\sum_{k=1}^{j} X_k \leq \xi_j,\ j = 1, \ldots, i\right\}$$

- Expressions can be accurately approximated recursively

$$G_i(\xi_1, \ldots, \xi_i) = P\left\{\sum_{k=1}^{j} X_k \leq \xi_j,\ j = 1, \ldots, i\right\}$$

Recursive computations

- Define random variables Y_i

$$P\{Y_1 \le x\} = P\{X_1 \le x\}$$

$$P\{Y_i \le x\} = \frac{G_i(\xi_1, \dots, \xi_{i-1}, x)}{G_{i-1}(\xi_1, \dots, \xi_{i-1})}, x \ge 0,\ i=2,\dots,N.$$

- Theorem

$$P\{Y_i \le x\} = \frac{P\{X_i + Y_{i-1} \le x, Y_{i-1} \le \xi_{i-1}\}}{P\{Y_{i-1} \le \xi_{i-1}\}}, i = 2, \dots, N$$

Two-moment recursion

- Fit mixture of Erlang distributions on $E[Y_i]$ and $\sigma^2(Y_i)$

$$E\left[Y_i\right] = E\left[X_i\right] + E\left[Y_{i-1} \middle| Y_{i-1} \le \xi_{i-1}\right],\ i = 2, \ldots, N\text{ -1}$$

$$\sigma^2\left(Y_i\right) = \sigma^2\left(X_i\right) + \sigma^2\left(Y_{i-1} \middle| Y_{i-1} \le \xi_{i-1}\right),\ i = 2, \ldots, N\text{ -1}$$

- Compute $P\{I_{ki} \geq 0\} = G_i(\xi_1, \ldots, \xi_i)$ recursively

$$G_1\left(\xi_1\right) = P\left\{Y_1 \le \xi_1\right\}$$

$$G_i\left(\xi_1, \ldots, \xi_i\right) = P\left\{Y_i \le \xi_i\right\} G_{i-1}\left(\xi_1, \ldots, \xi_{i-1}\right), i = 2, \ldots, N$$

Computational efficiency

- Solving cost-balance equations or generalized Newsvendor equations is equivalent to solving a one-dimensional bisection scheme to find a target value in (0,1)
- Finding base stock levels for N-item divergent structure requires solution of N bisection schemes
  - As efficient as solving N single-echelon systems
- Finding base stock levels for N-item general systems requires solution of at most 3N bisection schemes

Comparison against LP rolling schedule

LP-based rolling schedule approach

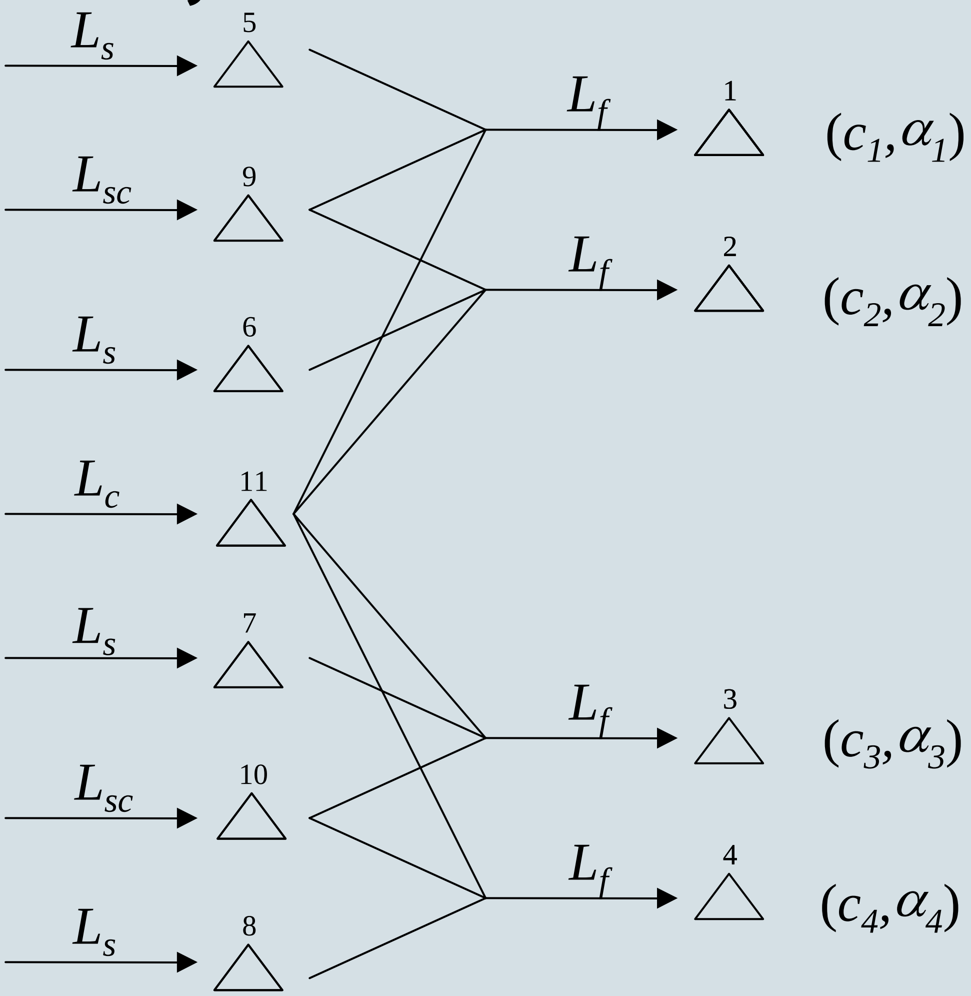
- Define objective function
  - linear holding and penalty costs
  - incorporate safety stocks by goal programming formulation
- Feasibility constraints
  - inventory balance equations at item level
  - material availability constraints w.r.t. dependent requirements
- No lot sizing constraints
- Solve LP problem
- Implement immediate decisions

Experimental comparison of SCOP paradigms: LP vs. SBS for uncapacitated systems

Example product structure

- 4 final products
  - E[D] = 100
  - P_1 = 95%
  - added value $10
  - assembly lead time 1
- 4 specific items
  - value $10
- 2 semi-specific items
  - value $30
- 1 common item
  - value $50

An 11-item system



Results simulation study

c_i^2	(L_f, L_s, L_{sc}, L_c)	Supply Chain Inventory capital				Customer service	
		SBS_{ana}	SBS_{sim}	LP	$\Delta\%$	$P_{1,SBS}$	$P_{1,LP}$
0.25	(1,1,2,4)	72188	71682	79477	11%	95%	95%
0.25	(1,4,2,1)	76154	76476	78133	2%	95%	95%
0.25	(1,1,4,2)	74162	73550	80620	10%	95%	95%
0.5	(1,1,2,4)	105114	104448	115227	10%	95%	95%
0.5	(1,4,2,1)	112226	112316	114659	2%	95%	95%
0.5	(1,1,4,2)	108079	107616	115386	7%	95%	95%
1	(1,1,2,4)	152583	152203	168533	11%	95%	95%
1	(1,4,2,1)	165264	165328	169122	2%	95%	95%
1	(1,1,4,2)	157294	157034	169030	8%	95%	95%
2	(1,1,2,4)	217664	218551	247578	13%	94%	95%
2	(1,4,2,1)	246637	245998	249849	2%	95%	95%
2	(1,1,4,2)	228967	228789	247551	8%	94%	95%

Impact of final product cost on safety stocks, LP

$(c_1^2, c_2^2, c_3^2, c_4^2)$	(h_1, h_2, h_3, h_4)	(L_f, L_s, L_{sc}, L_c)	LP-based safety stocks					
			P_1	P_2	ss_1	ss_2	ss_3	ss_4
(0.25, 0.25, 0.25, 0.25)	(20, 20, 20, **10**)	(1,1,4,2)	0.905	0.95	139	200	134	**234**
(0.25, 0.25, 0.25, 0.25)	(20, 20, 20, **10**)	(1,1,2,4)	0.905	0.95	131	162	137	**248**
(0.50, 0.50, 0.50, 0.50)	(20, 20, 20, **10**)	(1,1,2,4)	0.925	0.95	223	260	224	**373**
(0.50, 0.50, 0.50, 0.50)	(20, 20, 20, **10**)	(1,1,4,2)	0.925	0.95	247	307	220	**359**
(1, 1, 1, 1)	(20, 20, 20, **10**)	(1,1,2,4)	0.94	0.95	380	439	352	**541**
(1, 1, 1, 1)	(20, 20, 20, **10**)	(1,1,4,2)	0.94	0.95	416	475	346	**532**
(2, 2, 2, 2)	(20, 20, 20, **10**)	(1,1,2,4)	0.95	0.95	583	663	571	**812**
(2, 2, 2, 2)	(20, 20, **10, 10**)	(1,1,4,2)	0.945	0.95	613	664	**643**	**715**
(0.25, 0.25, **0.50, 0.50**)	(10, 10, 10, 10)	(1,1,4,2)	0.915	0.95	149	217	**289**	**260**

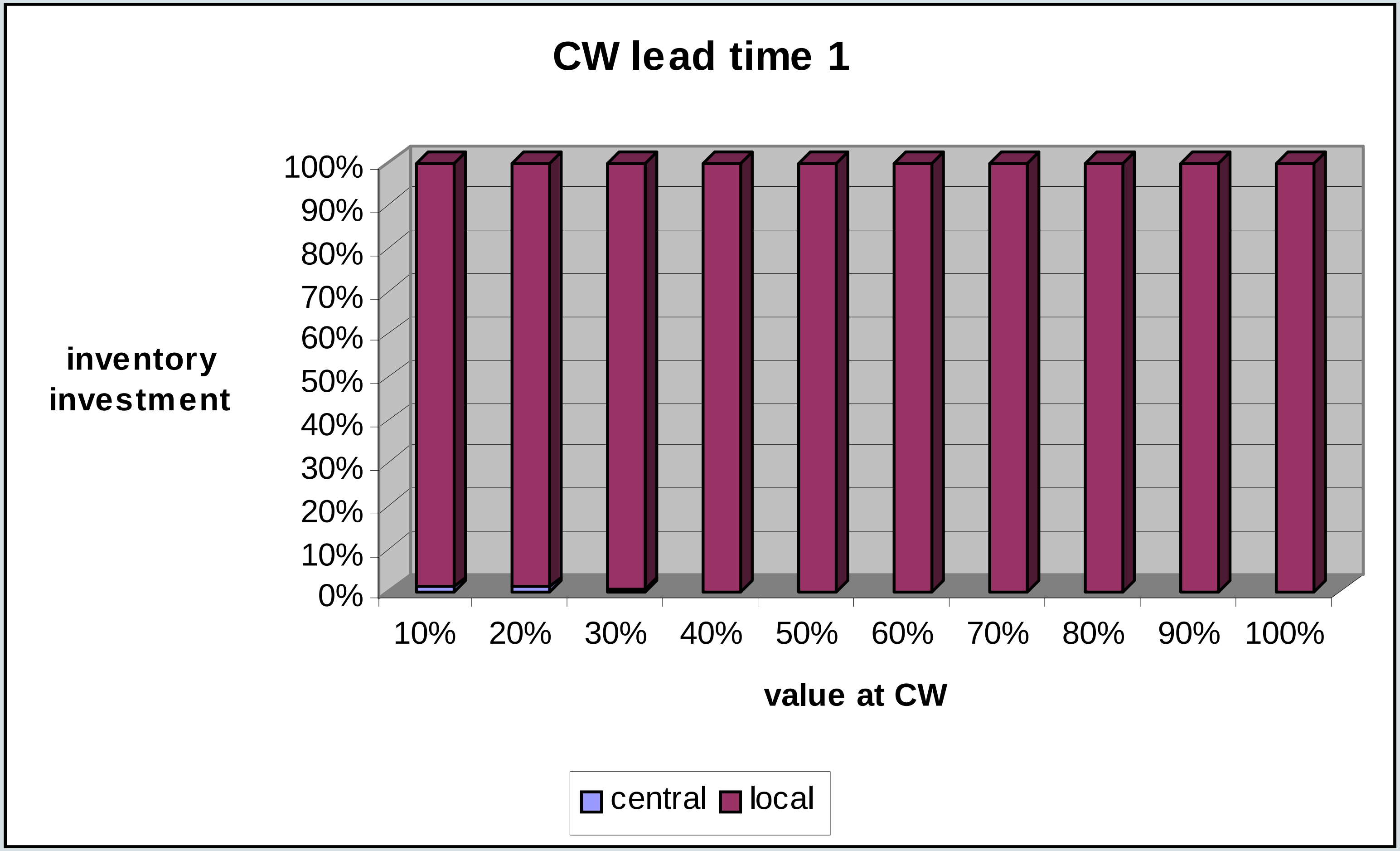
Impact of final product cost on safety stocks, SBS

$(c_1^2, c_2^2, c_3^2, c_4^2)$	(h_1, h_2, h_3, h_4)	(L_f, L_s, L_{sc}, L_c)	SBS-based safety stocks					
			P_1	P_2	ss_1	ss_2	ss_3	ss_4
(0.25, 0.25, 0.25, 0.25)	(20, 20, 20, **10**)	(1,1,4,2)	0.91	0.95	165	165	166	**189**
(0.25, 0.25, 0.25, 0.25)	(20, 20, 20, **10**)	(1,1,2,4)	0.91	0.95	165	165	166	**189**
(0.50, 0.50, 0.50, 0.50)	(20, 20, 20, **10**)	(1,1,2,4)	0.93	0.95	280	280	280	**319**
(0.50, 0.50, 0.50, 0.50)	(20, 20, 20, **10**)	(1,1,4,2)	0.93	0.95	280	280	281	**319**
(1, 1, 1, 1)	(20, 20, 20, **10**)	(1,1,2,4)	0.95	0.95	475	475	479	**538**
(1, 1, 1, 1)	(20, 20, 20, **10**)	(1,1,4,2)	0.95	0.95	475	475	477	**543**
(2, 2, 2, 2)	(20, 20, 20, **10**)	(1,1,2,4)	0.96	0.94	815	815	818	**911**
(2, 2, 2, 2)	(20, 20, **10**, **10**)	(1,1,4,2)	0.96	0.95	822	822	**842**	**842**
(0.25, 0.25, **0.50**, **0.50**)	(10, 10, 10, 10)	(1,1,4,2)	0.92	0.95	190	190	**319**	**319**

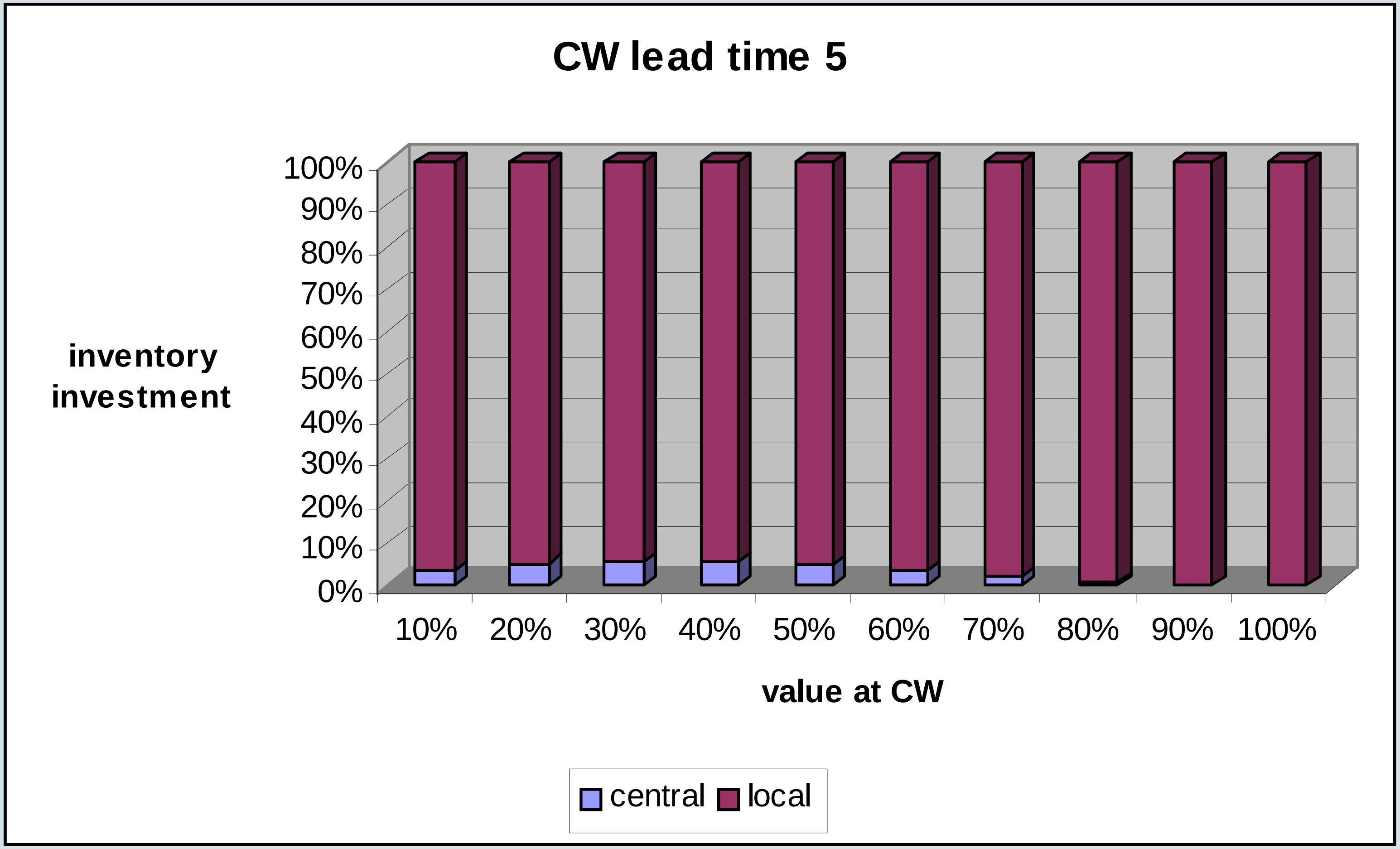
Managerial insights

- Most of inventory capital tied up downstream
  - informally speaking optimal upstream stock never exceeds 10% of overall supply chain inventory capital
- Flexibility only pays off downstream
  - lead time reduction
  - replenishment frequency
- Focus on efficiency upstream
- Postponement should come almost for free
- Shortages and overages should be shared among parent items
- Moving CODP upstream yields substantial savings
- Control parameters, i.e. safety stocks, depend on the operational planning logic applied

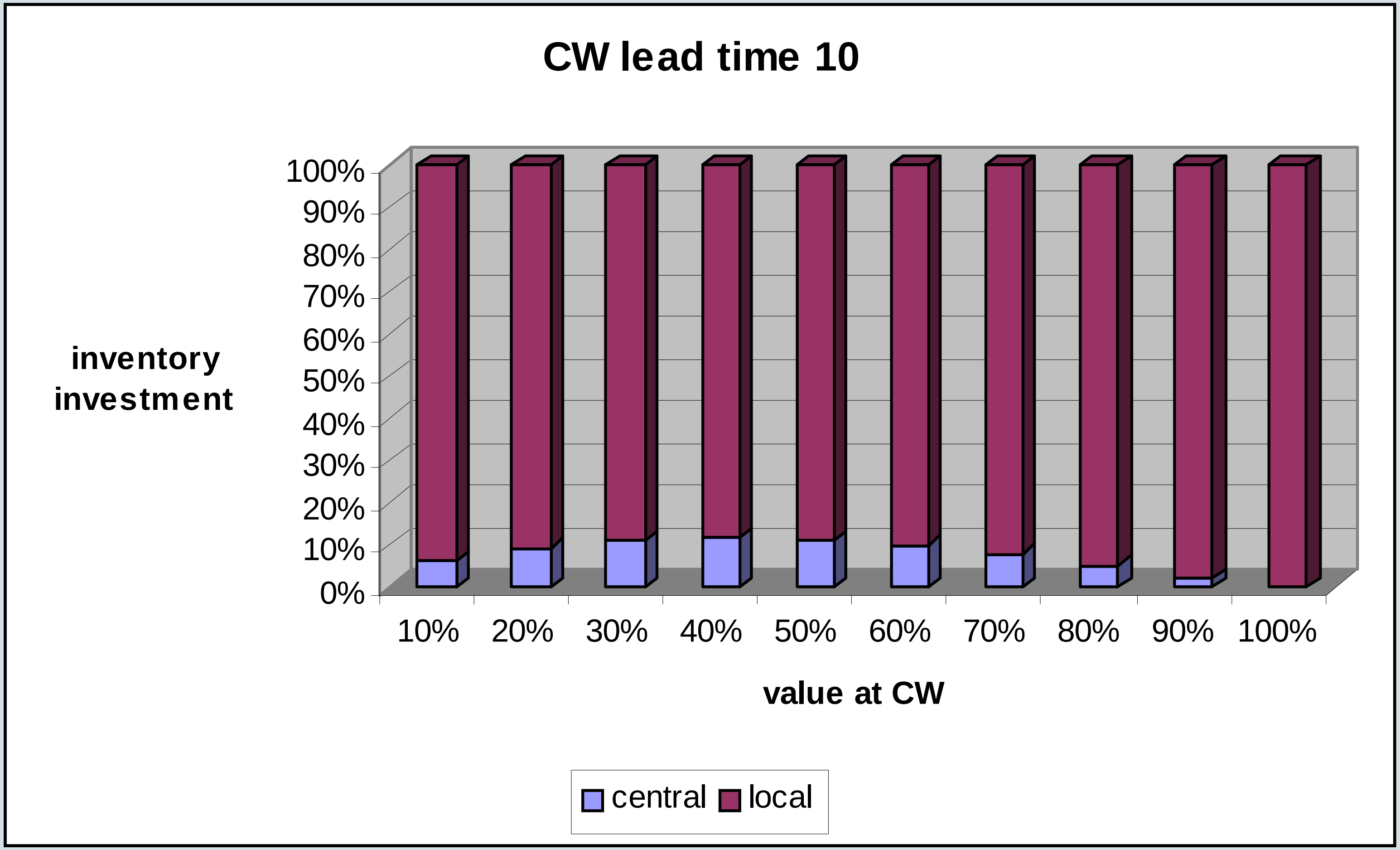
Inventory capital allocation



Inventory capital allocation



Inventory capital allocation



/faculteit technologie management

Lot sizing and allocation rules

- R_i review period of item i

$$r_j = \frac{R_i}{R_j}, r_j \in \mathbb{N}, j \in V_i$$

V_{it} set of successors of item i that order at time t

θ_{jt} the last ordering moment of successor j at or before time t

- Allocation policy should be such that net stock of i non-negative
  - Linear allocation policies defined before in general do not satisfy this constraint

Lot sizing and allocation rules

$$X_i(t) = \sum_{j \in V_i} q_j \left(\Delta_j - W_i(t)\right)^+$$

Then we can recursively define the random variables $W(t)$,

$$W(L_i) = Z_i(0) + D_i\left(0, L_i\right]$$
$$W(t) = \frac{1}{\sum_{j \in V_{it}} q_j} \left\{ Z_i(t - L_i) + D_i\left(t - L_i, t\right] - \sum_{j \notin V_{it}} D_j\left(\theta_{jt}, t\right] - \sum_{j \notin V_{it}} q_j W(\theta_{jt}) \right\}$$

The allocation policy follows from

$Y_j(t) = S_j - q_j\left(W(t) - \Delta_j\right)^+$, if j orders at time t

Conclusions

- SBS policies allow for control of general multi-item multi-echelon networks
- Close-to-optimal SBS policies can be efficiently computed
- SBS policies perform well compared with standard rolling scheduling
- SBS policies can be implemented for operational control
  - requires forecast-dependent base stock levels

Challenges for further research

- Extension of SBS with (period order) lot sizing
- Incorporation of multi-level forecasting
  - future detailed forecasts are conditional on today's aggregate forecast
  - multi-level forecast structure should be aligned with value network structure
- Finite capacity
  - Should finite capacity be modelled
    - explicitly through constraints or
    - explicitly through model of transformation process or
    - implicitly through planned lead time, i.e. loosely coupling SCOP-level and shopfloor control level
- Empirical research to test various modelling options