Optimal buffer allocation in serial production lines operating under Installation Buffer (IB), Echelon Buffer (EB), and CONWIP policies

George Liberopoulos
University of Thessaly
Department of Mechanical Engineering
Volos, Greece

Presentation at 11th SMMSO 2017: June 4-9, 2017 Acaya (Lecce), Italy

Outline

- Installation Buffer (IB) vs Echelon Buffer (EB) policy
- Performance evaluation of EB policy (sketch)
- Buffer Allocation Problem
- Solution methodology
- Numerical results on the optimal IB, EB, and CONWIP policies
- Conclusions

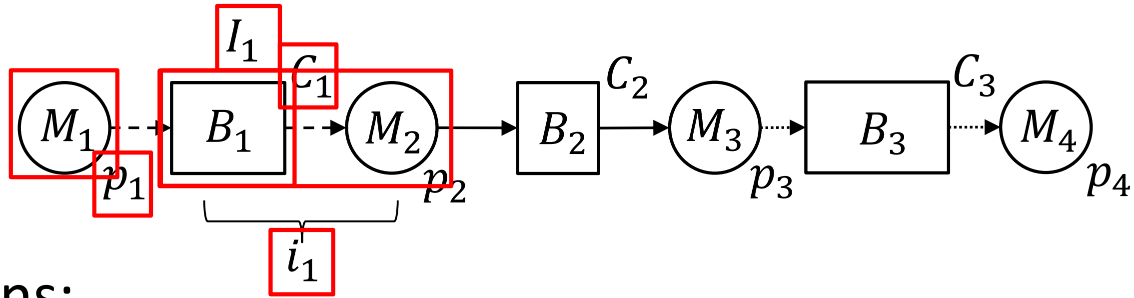
Installation Buffer (IB)

vs

Echelon Buffer (EB) policy

Installation Buffer (IB) policy

- Serial production line operated under IB policy



- Definitions:
  - M_n = machine, $n = 1, \dots, N$
  - p_n = probability that M_n produces a part in a period, $n = 1, \dots, N$
  - B_n = intermediate buffer, $n = 1, \dots, N-1$
  - C_n = intermediate buffer capacity, $n = 1, \dots, N-1$
  - I_n = installation buffer (IB), $n = 1, \dots, N-1$

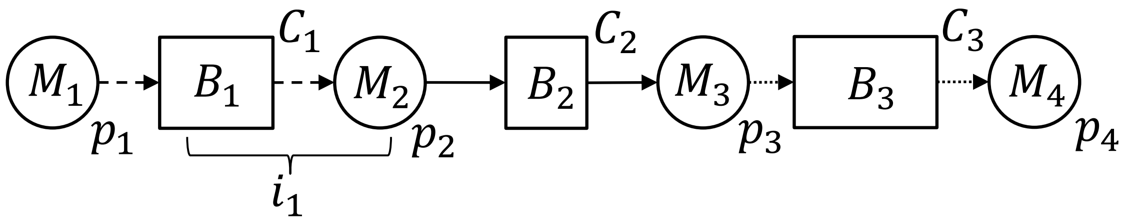
$$I_n = B_n \cup M_{n+1}, n = 1, \dots, N-1$$

  - i_n = installation WIP, $n = 1, \dots, N-1$

$$i_n \leq 1 + C_n \equiv \text{ capacity of } I_n$$

IB policy

- Serial production line operated under IB policy



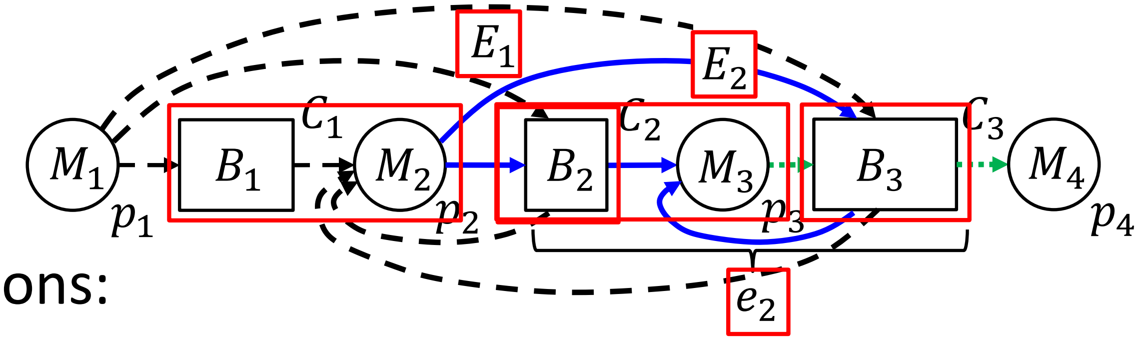
- Operation:
  - Machine M_n is allowed to store the parts that it produces in its immediate downstream buffer B_n if the next machine M_{n+1} is occupied, $n = 1, \dots, N - 1$.
  - Machine M_n is blocked before service from processing a part if the number of parts that have been produced by it but have not yet departed from the next machine M_{n+1} is equal to $1 + C_n$, i.e.,

$$i_n = 1 + C_n$$

Is there a way to increase the utilization of buffer space?

Echelon Buffer (EB) policy

- Serial production line operated under EB policy



- Definitions:
  - E_n = Echelon buffer (EB), $n = 1, \ldots, N-1$

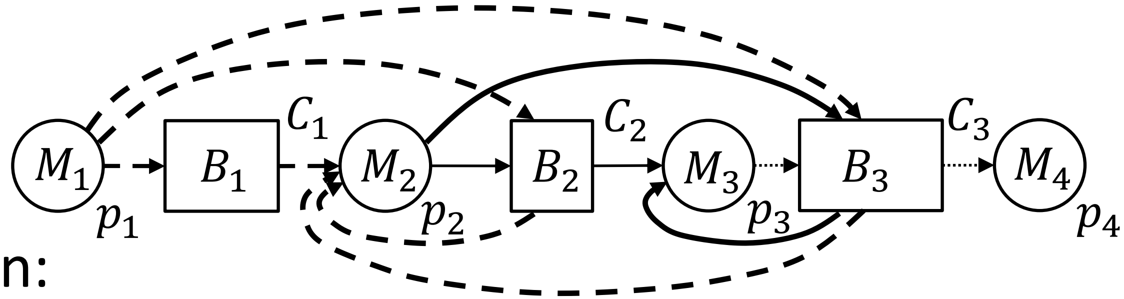
$$E_n = B_n \cup B_{n+1} \cup \cdots \cup B_{N-1} \cup M_{n+1}, n = 1, \ldots, N-1$$

  - e_n = echelon WIP, $n = 1, \ldots, N-1$

$$e_n \leq 1 + \sum_{m=n}^{N-1} C_m \equiv \text{ capacity of } E_n$$

EB policy

- Serial production line operated under EB policy

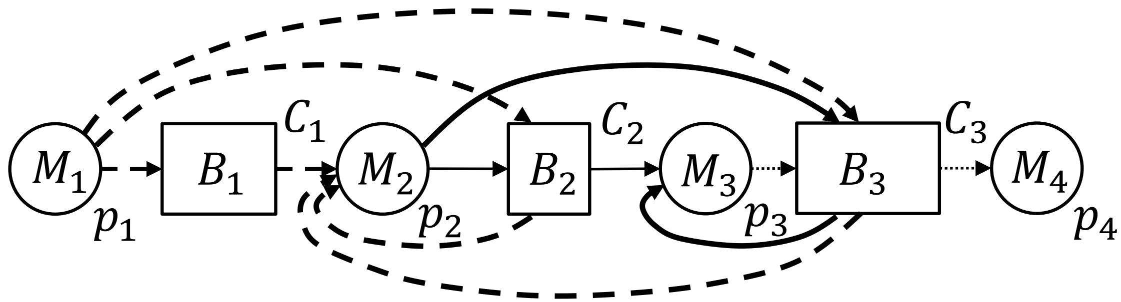


- Operation:
  - Machine M_n is allowed to store the parts that it produces in any of its downstream buffers $B_n, \dots, B_{N-1}$ if the next machine M_{n+1} is occupied, $n = 1, \dots, N-1$.

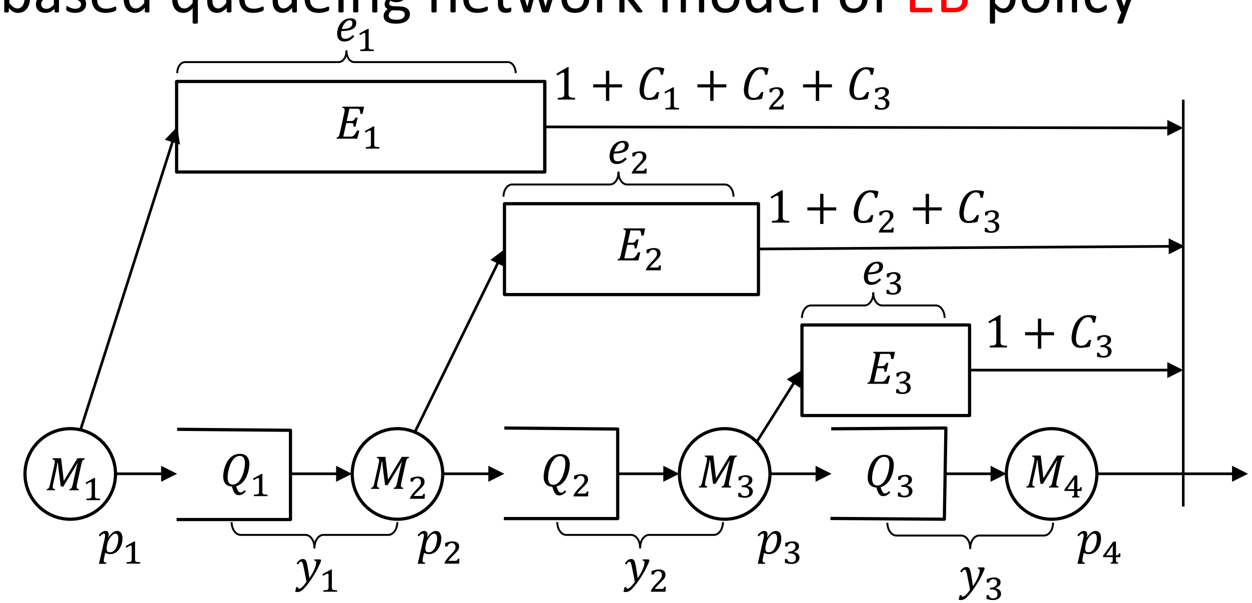
  - Machine M_n is blocked before service from processing a part if the number of parts that have been produced by it but have not yet departed from the last machine M_N is equal to $1 + \sum_{m=n}^{N-1} C_m$, i.e.,

$$e_n = 1 + \sum_{m=n}^{N-1} C_m$$

EB policy



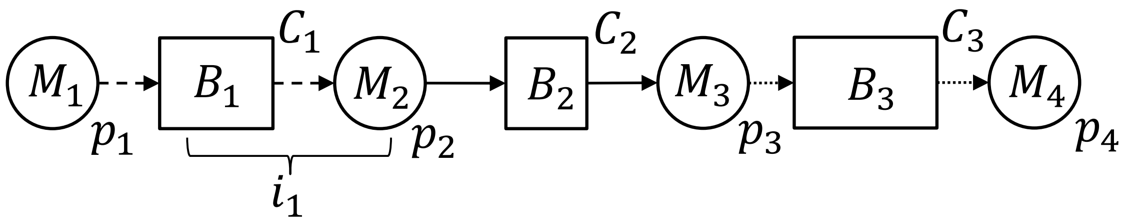
- Token-based queueing network model of EB policy



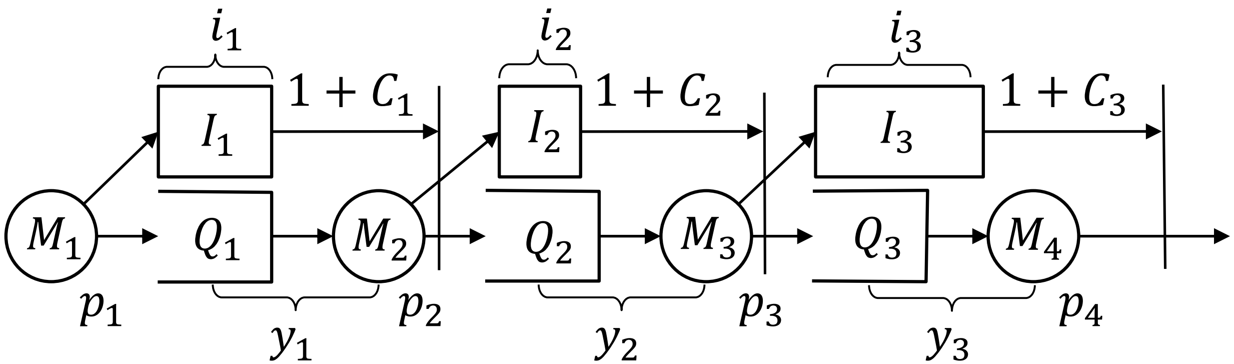
– $y_n \equiv$ stage WIP, $n = 1, \dots, N-1$

$$e_n = \sum_{m=n}^{N-1} y_m , n = 1, \dots, N-1$$

Back to IB policy



- Token-based queueing network model of IB policy

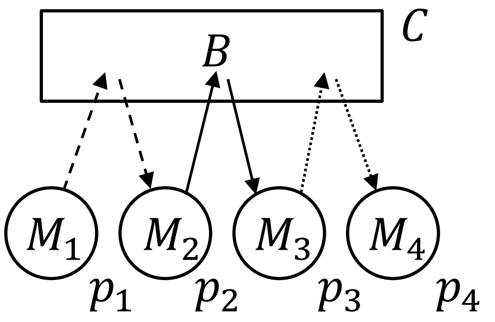


  - Q_n = stage-n queue, $n = 1, \dots, N-1$
  - y_n = stage WIP, $n = 1, \dots, N-1$

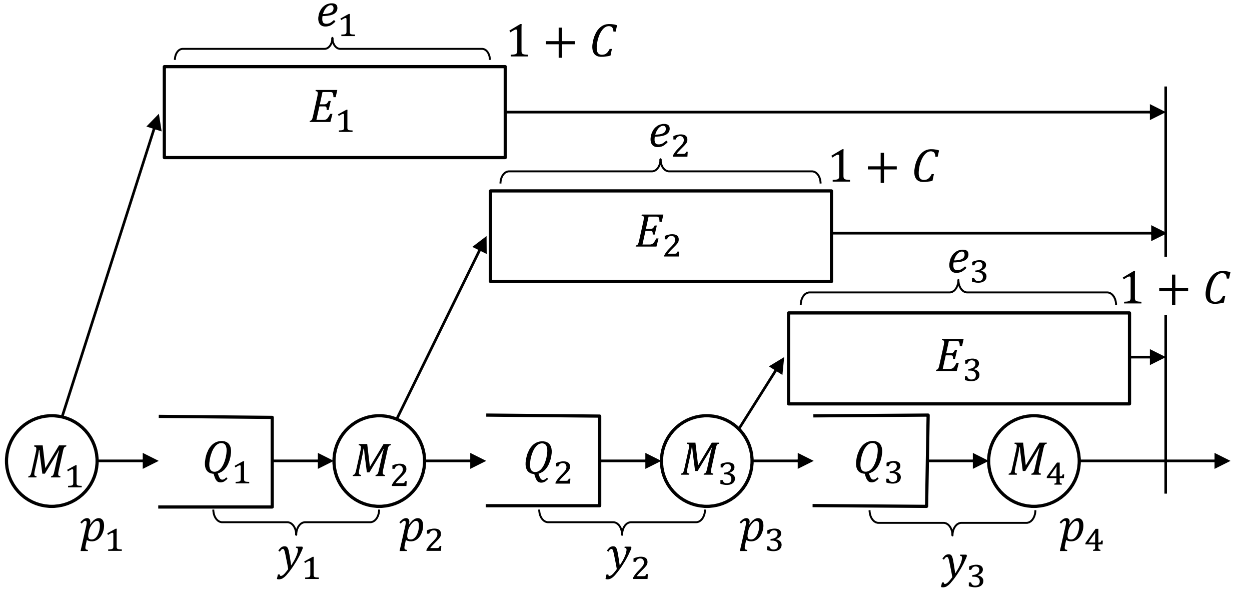
$$i_n = y_n, n = 1, \dots, N-1$$

EB policy special case

- $C_n = 0, n = 1, \ldots, N-2$ and $C_{N-1} = C \geq 0$: CONWIP



- Token-based queueing network model of EB policy



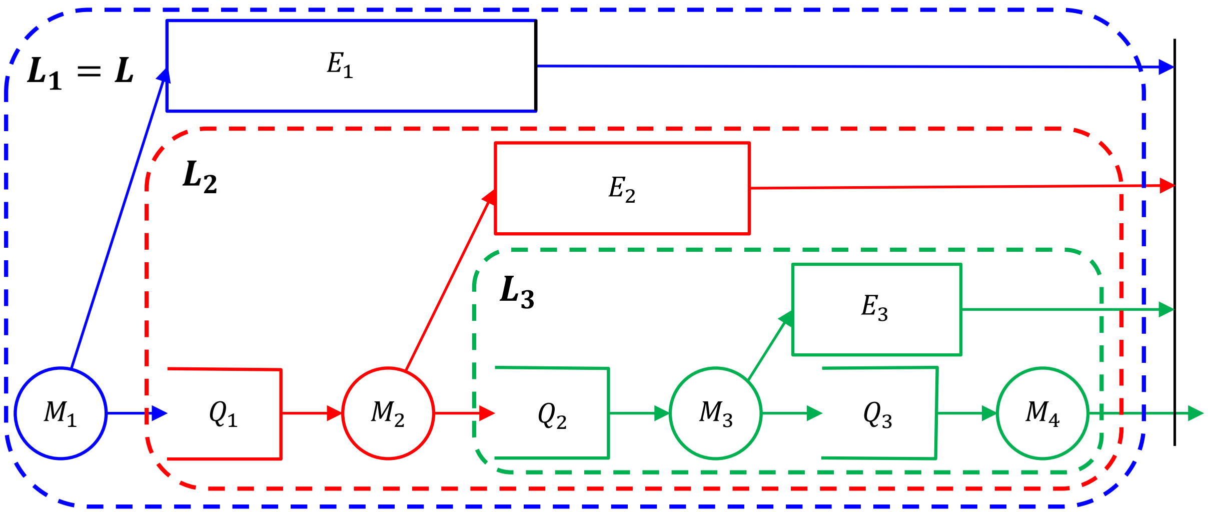
Advantages/disadvantages of EB

- Advantages of EB policy
  - Better utilization of buffer space
  - Global information
  - Fewer blockages and starvations of machines
  - Higher average throughput
- Disadvantages of EB policy
  - Higher average WIP
  - Higher transportation cost of parts to remote buffers
- To evaluate the advantages and disadvantages, need to:
  - Evaluate the performance of the EB policy
  - Optimize the parameters (buffer capacities) of the EB policy
  - Compare the optimal EB policy against the optimal IB and CONWIP policies

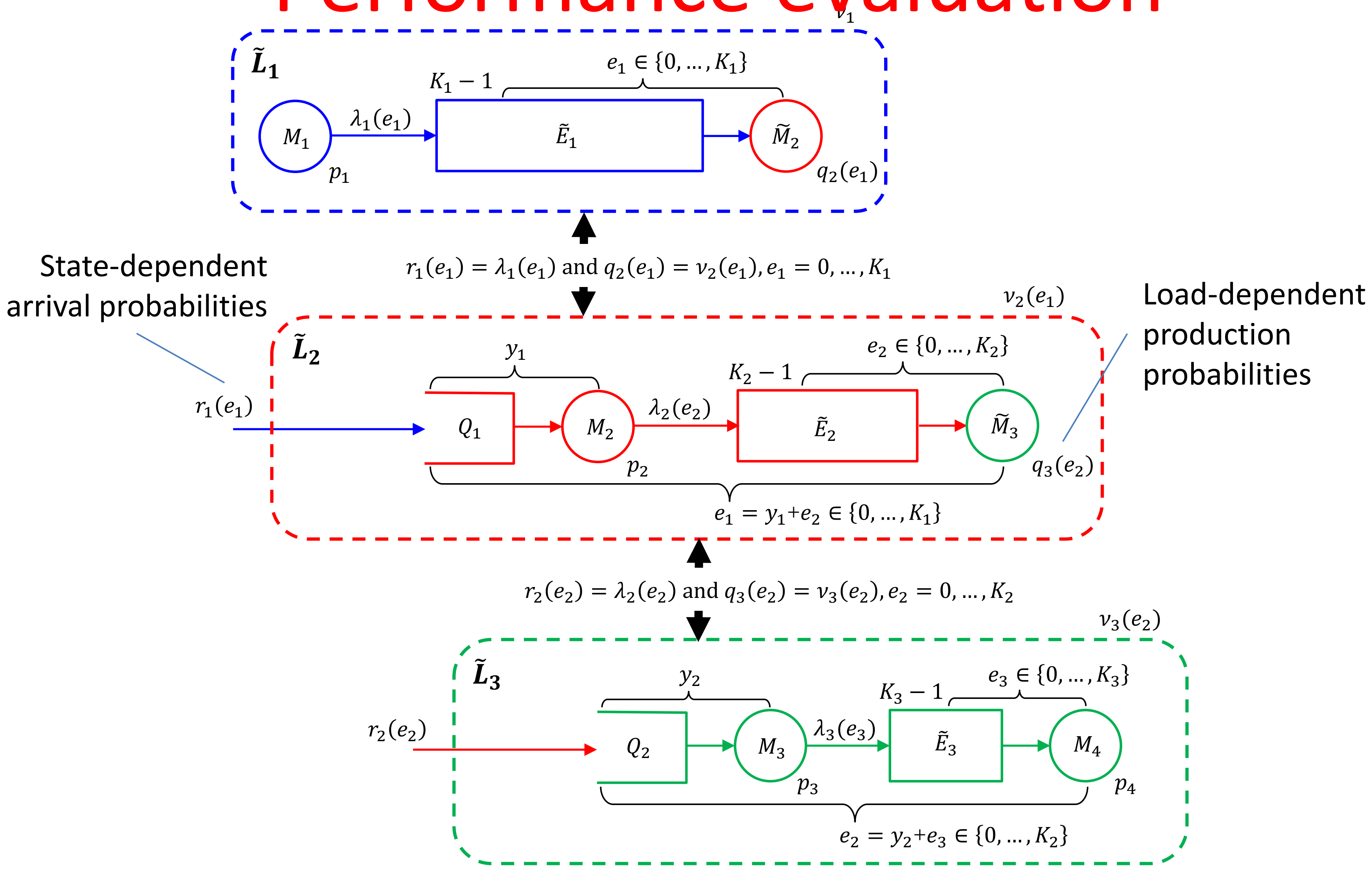
Performance evaluation

- Decomposition-based approximation method:
  1. Decompose the original queueing network of N machines and $N-1$ buffers into $N-1$ nested segments.
  2. Approximate each segment with a 2-machine subsystem than can be analyzed in isolation as a 2D MC.
  3. Determine the parameters of the 2-machine subsystems by relationships among the flows of parts in the original system.
- The approximation method is highly accurate and computationally efficient.

Performance evaluation



Performance evaluation



Buffer Allocation Problem (BAP)

Optimization problem (Shi and Gershwin 2009 adaptation):

Average net profit

Total buffer capacity

Overflow rate of stage WIP buffer Y_n

Average stage WIP

$$\max_{C_1,\ldots,C_{N-1}} P(C_1,\ldots,C_{N-1}) =$$

$$r\,\nu(C_1,\ldots,C_{N-1}) - \left(b\sum_{n=1}^{N-1} C_n + \sum_{n=1}^{N-1} h_n \bar{y}_n(C_1,\ldots,C_{N-1}) + t\sum_{n=1}^{N-1} \theta_n(C_1,\ldots,C_{N-1}) \right)$$

Average throughput

$$\text{s.t.} \quad \nu(C_1,\ldots,C_{N-1}) \geq \nu_{min}$$
$$C_n \geq 0, n = 1,\ldots,N-1$$

θ_n = Prob [M_n produces a part AND $y_n \geq 1 + C_n$]

r = gross marginal profit (€ per part produced)

b = storage space cost (€ per storage slot per unit time)

h_n = stage WIP inventory holding cost (€ per part in y_n per unit time)

t = cost rate of transferring parts to remote buffers (€ per part transferred);

ν_{min} = minimum required average throughput

BAP: Solution methodology

Solution methodology (Shi and Gershwin 2009 adaptation):

1. Solve unconstrained problem (using "gradient" search: step by step increments)

If $v(C_1^*, \ldots, C_{N-1}^*) \geq v_{min}$ then GOTO EXIT

2. (Else) set $r = r + \Delta r$ and resolve unconstrained problem (Lagrange multiplier method)

If $v(C_1^*, \ldots, C_{N-1}^*) \geq v_{min}$ then GOTO EXIT

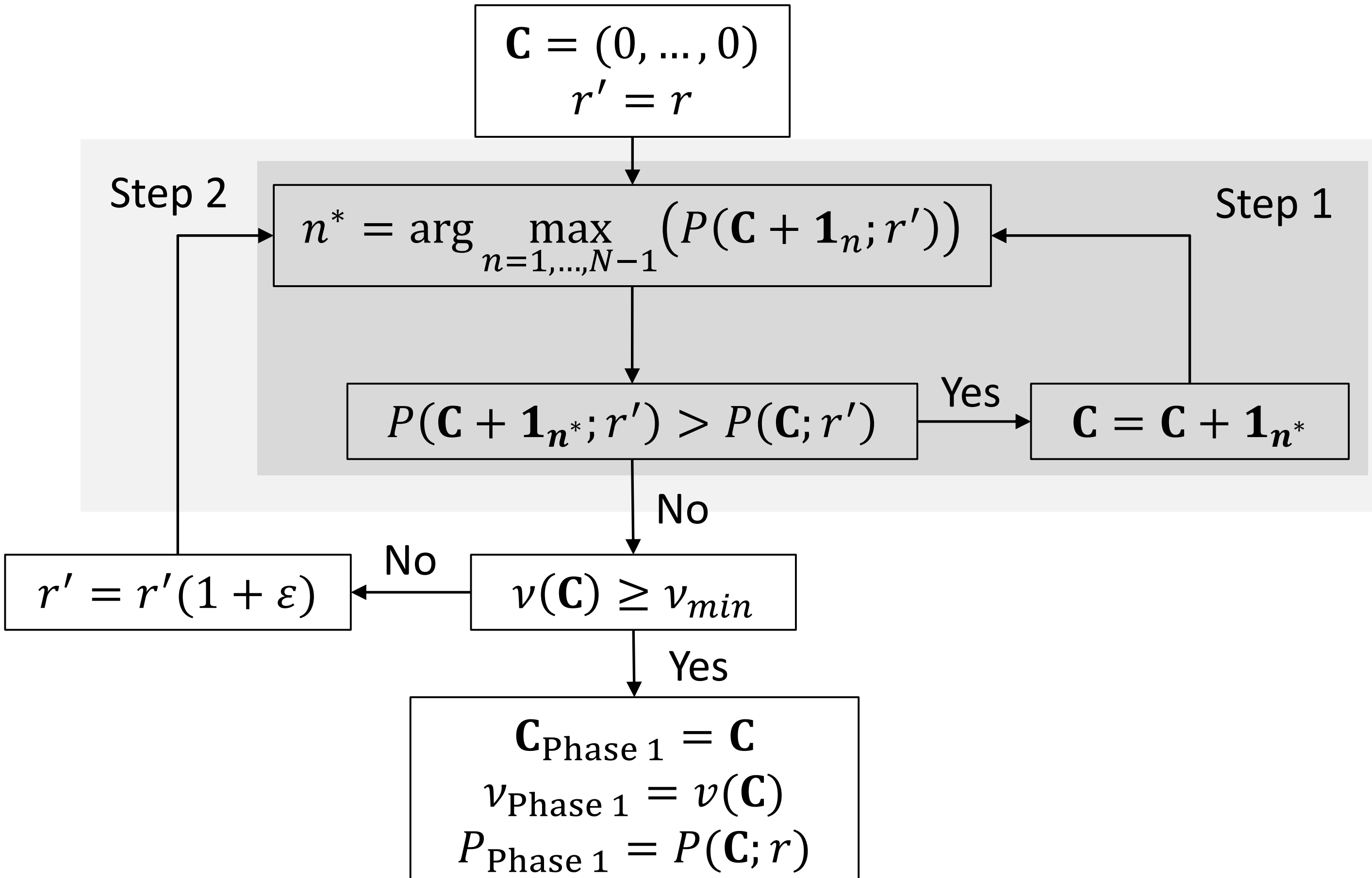
Else GOTO step 2

EXIT: Compute final maximum average profit using original value of r

(Phase 2: perform a local search to improve solution)

BAP: Solution methodology

- Solution methodology (Shi and Gershwin 2009 adaptation):



Numerical results: Input parameters

- Experimental design:
  - To systematically compare the IB, EB, and CONWIP policies, we optimized each policy for an $N = 8$ machine line and several scenarios for the
    - Production rates $p_n, n = 1, \ldots., N$
    - Parameters $r, b, h_n, n = 1, \ldots, N-1, v_{min}$
    - Parameter t

Numerical results: Input parameters

- Production rate scenarios

#	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8
L_1	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6
L_2	0.6	**0.5**	0.6	0.6	0.6	0.6	0.6	0.6
L_3	0.6	0.6	0.6	0.6	**0.5**	0.6	0.6	0.6
L_4	0.6	0.6	0.6	0.6	0.6	0.6	**0.5**	0.6
L_5	0.53	0.55	0.57	0.59	0.61	0.63	0.65	0.67
L_6	0.67	0.65	0.63	0.61	0.59	0.57	0.55	0.53

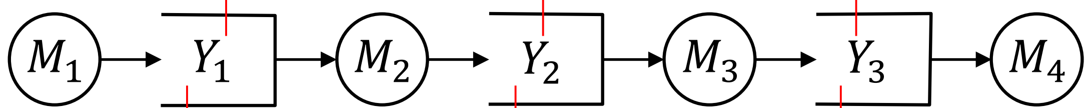
Numerical results: Input parameters

Scenarios for parameters $r, b, h_n, n = 1, \dots, N-1, v_{min}$: Auxiliary input parameters

$I_c = 0.5$

$c_0 = 100$ $c_1 = 150$ $c_2 = 200$ $c_3 = 250$ $c_4 = 300$

$I_r = 0.1$



$r = I_r c_N = 30$

$I_h = 0.001$

$h_1 = 0.15$ $h_2 = 0.2$ $h_3 = 0.25$

$I_b = 2$

$b = I_b h_1 = 0.3$

$I_v = 0.8$

$v_{min} = I_v p_{min} = 0.8\, p_{min}$

c_0 = raw-material cost (€ per raw part)

I_c = value-added rate per production stage as a percentage of c_0

c_n = total cost per part produced by M_n (€ per part) $c_n = c_{n-1} + I_c c_0 = c_0(1 + nI_c)$

I_h = interest rate (€ per € invested per unit time) $h_n = I_h c_n$

I_r = gross profit margin $r = I_r c_N$

I_b = buffer space cost multiplier w.r.t. h_1 $b = I_b h_1$

I_v = minimum required production line efficiency $v_{min} = I_v p_{min}$

Numerical results: Input parameters

Scenarios for parameters $r, b, h_n, n = 1, \dots, N-1$, based on auxiliary input parameters

	Auxiliary parameters						Optimization parameters								
#	c_0	I_c	I_h	I_r	I_b	I_v	r	b	h_1	h_2	h_3	h_4	h_5	h_6	h_7
O_1	100	0.5	0.001	0.10	2	0.8	50	0.30	0.15	0.2	0.25	0.3	0.35	0.4	0.45
O_2	100	**0.0**	0.001	0.10	2	0.8	10	0.20	0.10	0.1	0.10	0.1	0.10	0.1	0.10
O_3	100	**1.0**	0.001	0.10	2	0.8	90	0.40	0.20	0.3	0.40	0.5	0.60	0.7	0.80
O_4	100	**5.0**	0.001	0.10	2	0.8	410	1.20	0.60	1.1	1.60	2.1	2.60	3.1	3.60
O_5	100	**10.0**	0.001	0.10	2	0.8	810	2.20	1.10	2.1	3.10	4.1	5.10	6.1	7.10
O_6	100	0.5	**0.002**	0.10	2	0.8	50	0.60	0.30	0.4	0.50	0.6	0.70	0.8	0.90
O_7	100	0.5	0.001	**0.20**	2	0.8	100	0.30	0.15	0.2	0.25	0.3	0.35	0.4	0.45
O_8	100	0.5	0.001	**0.05**	2	0.8	25	0.30	0.15	0.2	0.25	0.3	0.35	0.4	0.45
O_9	100	0.5	0.001	0.10	**0**	0.8	50	0.00	0.15	0.2	0.25	0.3	0.35	0.4	0.45
O_{10}	100	0.5	0.001	0.10	**1**	0.8	50	0.15	0.15	0.2	0.25	0.3	0.35	0.4	0.45
O_{11}	100	0.5	0.001	0.10	2	**0.0**	50	0.30	0.15	0.2	0.25	0.3	0.35	0.4	0.45

Numerical results: Input parameters

- Scenarios for transfer cost rate t

I_t = transfer cost rate as multiple of raw material cost c_0 $t = I_t c_0$

$c_0 = 100$ $I_t = 0.01$ $t = 1$

#	I_t		t
T_1	0.00		0
T_2	0.01		1
T_3	0.02		2

Numerical results: Experiments

- We run all combinations of scenarios $L_1 – L_6, T_1 – T_3, O_1 – O_{11}$
- Total number of instances: $6 \times 3 \times 11 = 198$
- In each instance, we optimized the installation buffer capacities for:
  - Installation buffer (IB) policy $(C_1, \dots, C_7)$
  - Echelon buffer (EB) policy $(C_1, \dots, C_7)$
  - CONWIP policy $(C \equiv C_7;\ C_1 = \cdots = C_6 = 0)$

Numerical results: Optimal values

- Results for scenarios $L_1, T_1, O_1 - O_{11}$ (balanced line, $t = 0$)

		IB Policy			EB Policy			CONWIP Policy			$\%\Delta P^*$	
	ν_{min}	$[C_1^*$-$C_7^*]$	ν^*	P^*	$[C_1^*$-$C_7^*]$	ν^*	P^*	C^*	ν^*	P^*	E-I	E-C
O_1	0.480	[2,3,4,3,4,3,3]	0.480	13.465	[0,0,0,1,0,5,8]	0.483	15.991	14	0.483	15.991	18.757	0.0009
O_2	0.480	[2,3,4,3,4,3,3]	0.480	-0.926	[0,0,0,0,0,5,9]	0.483	0.713	14	0.483	0.713	176.979	0.0011
O_3	0.480	[2,3,4,3,4,3,3]	0.480	27.857	[0,0,0,1,0,5,8]	0.483	31.268	14	0.483	31.268	12.247	0.0009
O_4	0.480	[2,3,4,3,4,3,3]	0.480	142.989	[0,0,0,1,0,6,7]	0.483	153.490	14	0.483	153.488	7.344	0.0011
O_5	0.480	[3,3,3,4,3,4,3]	0.485	286.147	[0,0,1,0,2,4,8]	0.490	306.436	15	0.490	306.431	7.090	0.0015
O_6	0.480	[2,3,4,3,4,3,3]	0.480	2.915	[0,0,1,2,1,3,7]	0.483	7.854	14	0.483	7.853	169.458	0.0098
O_7	0.480	[2,3,4,3,4,3,3]	0.480	37.481	[0,0,0,0,2,5,10]	0.502	40.400	17	0.502	40.400	7.789	0.0004
O_8	0.480	[2,3,4,3,4,3,3]	0.480	1.457	[0,0,1,2,1,3,7]	0.483	3.927	14	0.483	3.927	169.458	0.0098
O_9	0.480	[2,5,5,5,5,5,5]	0.502	20.460	[0,0,0,1,1,6,9]	0.502	20.388	17	0.502	20.388	-0.352	0.0010
O_{10}	0.480	[2,3,4,3,4,3,3]	0.480	16.765	[0,0,0,1,0,5,8]	0.483	18.091	14	0.483	18.091	7.907	0.0008
O_{11}	0.000	[1,2,2,2,2,2,2]	0.421	14.515	[0,0,0,0,1,3,7]	0.454	16.234	11	0.454	16.234	11.843	0.0004

Numerical results: Optimal values

- Results for scenarios $L_1, T_2, O_1 - O_{11}$ (balanced line, $t = 0.01$)

		IB Policy			EB Policy			CONWIP Policy			$\%\Delta P^*$	
	v_{min}	$[C_1^*$-$C_7^*]$	v^*	P^*	$[C_1^*$-$C_7^*]$	v^*	P^*	C^*	v^*	P^*	E-I	E-C
O_1	0.480	[2,3,4,3,4,3,3]	0.480	13.465	[0,1,1,2,2,3,5]	0.482	15.504	14	0.483	15.067	15.142	2.8999
O_2	0.480	[2,3,4,3,4,3,3]	0.480	-0.926	[0,1,2,2,2,3,4]	0.481	0.268	14	0.483	-0.210	128.876	227.1177
O_3	0.480	[2,3,4,3,4,3,3]	0.480	27.857	[0,1,1,2,2,3,5]	0.482	30.767	14	0.483	30.345	10.449	1.3935
O_4	0.480	[2,3,4,3,4,3,3]	0.480	142.989	[0,1,1,1,2,3,6]	0.482	152.942	14	0.483	152.564	6.961	0.2477
O_5	0.480	[3,3,3,4,3,4,3]	0.485	286.147	[0,0,1,2,2,3,7]	0.490	305.865	15	0.490	305.478	6.891	0.1265
O_6	0.480	[2,3,4,3,4,3,3]	0.480	2.915	[1,1,1,2,2,2,5]	0.480	7.391	14	0.483	6.930	153.565	6.6543
O_7	0.480	[2,3,4,3,4,3,3]	0.480	37.481	[0,1,2,2,2,4,6]	0.502	39.850	16	0.496	39.401	6.322	1.1394
O_8	0.480	[2,3,4,3,4,3,3]	0.480	1.457	[1,1,1,2,2,2,5]	0.480	3.482	14	0.483	3.003	138.934	15.9541
O_9	0.480	[2,5,5,5,5,5,5]	0.502	20.460	[1,1,2,2,2,3,6]	0.501	19.880	17	0.502	19.384	-2.832	2.5595
O_{10}	0.480	[2,3,4,3,4,3,3]	0.480	16.765	[0,1,1,2,2,3,5]	0.482	17.604	14	0.483	17.167	5.003	2.5451
O_{11}	0.000	[1,2,2,2,2,2,2]	0.421	14.515	[0,1,1,1,2,2,4]	0.453	15.793	11	0.454	15.423	8.802	2.3997

Numerical results: Optimal values

- Results for scenarios $L_1, T_3, O_1 - O_{11}$ (balanced line, $t = 0.02$)

		IB Policy			EB Policy			CONWIP Policy			$\%\Delta P^*$	
	v_{min}	$[C_1^*\text{-}C_7^*]$	v^*	P^*	$[C_1^*\text{-}C_7^*]$	v^*	P^*	C^*	v^*	P^*	E-I	E-C
O_1	0.480	[2,3,4,3,4,3,3]	0.480	13.465	[1,1,1,2,2,2,5]	0.480	15.064	14	0.483	14.143	11.874	6.5083
O_2	0.480	[2,3,4,3,4,3,3]	0.480	-0.926	[0,1,2,2,2,3,4]	0.481	-0.159	14	0.483	-1.134	82.846	85.9868
O_3	0.480	[2,3,4,3,4,3,3]	0.480	27.857	[0,1,1,2,2,3,5]	0.482	30.301	14	0.483	29.421	8.775	2.9910
O_4	0.480	[2,3,4,3,4,3,3]	0.480	142.989	[0,1,1,1,2,3,6]	0.482	152.437	14	0.483	151.641	6.607	0.5248
O_5	0.480	[3,3,3,4,3,4,3]	0.485	286.147	[0,1,1,1,2,3,7]	0.490	305.323	15	0.490	304.525	6.702	0.2619
O_6	0.480	[2,3,4,3,4,3,3]	0.480	2.915	[1,1,1,2,2,2,5]	0.480	6.964	14	0.483	6.006	138.934	15.9541
O_7	0.480	[2,3,4,3,4,3,3]	0.480	37.481	[1,1,1,2,3,3,6]	0.501	39.351	16	0.496	38.422	4.991	2.4192
O_8	0.480	[2,3,4,3,4,3,3]	0.480	1.457	[1,1,1,2,2,2,5]	0.480	3.056	14	0.483	2.079	109.670	46.9451
O_9	0.480	[2,5,5,5,5,5,5]	0.502	20.460	[1,1,2,2,3,3,5]	0.500	19.436	16	0.496	18.400	-5.005	5.6281
O_{10}	0.480	[2,3,4,3,4,3,3]	0.480	16.765	[1,1,1,2,2,2,5]	0.480	17.164	14	0.483	16.243	2.379	5.6669
O_{11}	0.000	[1,2,2,2,2,2,2]	0.421	14.515	[0,1,1,1,2,2,4]	0.453	15.387	10	0.441	14.631	6.004	5.1630

Numerical results: Optimal values

- Results for scenarios $L_2, T_2, O_1 - O_{11}$ (bottleneck upstream, $t = 0.01$)

		IB Policy			EB Policy			CONWIP Policy			$\%\Delta P^*$	
	v_{min}	$[C_1^*\text{-}C_7^*]$	v^*	P^*	$[C_1^*\text{-}C_7^*]$	v^*	P^*	C^*	v^*	P^*	E-I	E-C
O_1	0.400	[2,3,2,2,2,2,2]	0.424	13.829	[0,1,1,1,2,2,4]	0.433	14.951	10	0.422	14.584	8.115	2.5177
O_2	0.400	[2,2,2,2,2,2,1]	0.409	0.513	[0,1,1,1,2,2,2]	0.406	1.049	9	0.409	0.694	104.365	51.0399
O_3	0.400	[2,3,2,2,2,2,2]	0.424	27.434	[0,1,1,1,2,3,4]	0.443	29.329	12	0.444	28.943	6.909	1.3347
O_4	0.400	[2,3,3,3,2,2,2]	0.435	136.946	[0,1,1,1,2,3,6]	0.459	145.855	14	0.459	145.452	6.505	0.2770
O_5	0.400	[3,4,3,3,3,3,2]	0.457	276.487	[0,1,1,1,2,3,6]	0.459	291.752	14	0.459	291.358	5.521	0.1354
O_6	0.400	[2,2,2,2,2,2,1]	0.409	7.097	[0,1,1,1,1,2,3]	0.408	9.573	9	0.409	9.259	34.891	3.3905
O_7	0.400	[3,4,3,3,3,3,2]	0.457	35.809	[0,1,2,2,2,3,5]	0.465	37.605	15	0.465	37.150	5.014	1.2237
O_8	0.400	[2,2,2,2,2,2,1]	0.409	3.548	[0,1,1,1,1,2,3]	0.408	4.608	9	0.409	4.275	29.868	7.7805
O_9	0.400	[4,11,5,5,4,4,4]	0.483	19.821	[0,2,2,2,3,3,5]	0.474	18.921	16	0.470	18.410	-4.542	2.7726
O_{10}	0.400	[2,3,3,2,3,2,2]	0.434	16.146	[0,1,1,2,2,3,4]	0.452	16.718	12	0.444	16.291	3.541	2.6241
O_{11}	0.000	[1,2,2,2,1,2,1]	0.384	13.552	[0,1,1,1,2,2,4]	0.433	14.951	10	0.422	14.584	10.328	2.5177

Numerical results: Optimal values

- Results for scenarios $L_4, T_2, O_1 - O_{11}$ (bottleneck downstream, $t = 0.01$)

		IB Policy			EB Policy			CONWIP Policy			$\%\Delta P^*$	
	v_{min}	$[C_1^*\text{-}C_7^*]$	v^*	P^*	$[C_1^*\text{-}C_7^*]$	v^*	P^*	C^*	v^*	P^*	E-I	E-C
O_1	0.400	[1,2,2,2,2,3,2]	0.417	13.557	[0,1,1,1,1,2,4]	0.421	14.650	10	0.422	14.337	8.058	2.1796
O_2	0.400	[1,1,3,2,2,2,2]	0.401	0.500	[0,1,1,1,1,2,3]	0.406	1.037	9	0.409	0.694	107.437	49.3610
O_3	0.400	[1,2,2,2,2,3,2]	0.417	26.758	[0,1,1,1,1,2,5]	0.433	28.599	11	0.434	28.284	6.882	1.1153
O_4	0.400	[2,2,2,2,3,3,3]	0.436	133.684	[0,1,1,1,1,2,7]	0.452	141.276	12	0.444	140.960	5.679	0.2242
O_5	0.400	[2,2,2,2,3,3,3]	0.436	267.597	[0,0,1,1,2,2,7]	0.452	282.420	13	0.452	282.108	5.539	0.1107
O_6	0.400	[1,2,2,1,2,3,2]	0.401	6.557	[0,0,1,1,1,2,4]	0.408	9.146	9	0.409	8.858	39.487	3.2533
O_7	0.400	[2,2,2,3,3,4,3]	0.447	35.154	[0,1,1,1,2,3,6]	0.459	36.953	14	0.459	36.588	5.118	0.9978
O_8	0.400	[1,2,2,1,2,3,2]	0.401	3.278	[0,1,1,1,1,2,3]	0.406	4.399	9	0.409	4.075	34.180	7.9516
O_9	0.400	[1,5,4,4,4,4,9]	0.461	18.852	[1,1,1,1,2,3,5]	0.457	18.239	14	0.459	17.833	-3.251	2.2752
O_{10}	0.400	[1,2,2,2,2,3,3]	0.420	15.716	[0,1,1,1,1,3,5]	0.443	16.294	12	0.444	15.930	3.678	2.2836
O_{11}	0.000	[1,1,2,2,2,2,2]	0.394	13.516	[0,1,1,1,1,2,4]	0.421	14.650	10	0.422	14.337	8.389	2.1796

Conclusions

- We performed an extended numerical study comparing the optimal IB, EB and CONWIP policies
- EB significantly outperforms IB ($t = 0$)
- EB barely outperforms CONWIP ($t = 0$)
- As t ↗, dominance of EB over IB ↘
- As t ↗, dominance of EB over CONWIP ↗
- Total buffer capacity: CONWIP $= (<)$ EB << IB
- Numerically verified that the objective function of the EB policy is concave

Direction for future research

- Consider more complex machine models than the Bernoulli model

Acknowledgement

This research was supported by the ECSEL Joint Undertaking under grant agreement No 737459. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation program and Germany, Austria, France, Czech Republic, Netherlands, Belgium, Spain, Greece, Sweden, Italy, Ireland, Poland, Hungary, Portugal, Denmark, Finland, Luxembourg, Norway, Turkey.

Thank you for your attention.

Any questions?