SMMSO 2017

Vehicle Configuration in an Integrated Material Handling System with State-dependent Batch Transfer

Hui-Yu ZHANG; Shao-Hui XI; Qing-Xin CHEN; Xiang LI; Ai-Lin YU

Guangdong University of Technology

Department of Industrial Engineering

Guangzhou, CHINA

June 12, 2017

Outline

1 Introduction

- In many manufacturing environments, **production systems** (workstations) are interconnected with **material handling systems** (e.g. automated guided vehicles).

- Performance modeling of such integrated systems is a complex problem because of the **state-dependency** on the arrival and departure processes.

- Estimating **the number of vehicles required** is one of the most important issues at the tactical level of the design and control processes of such systems.

1 Introduction

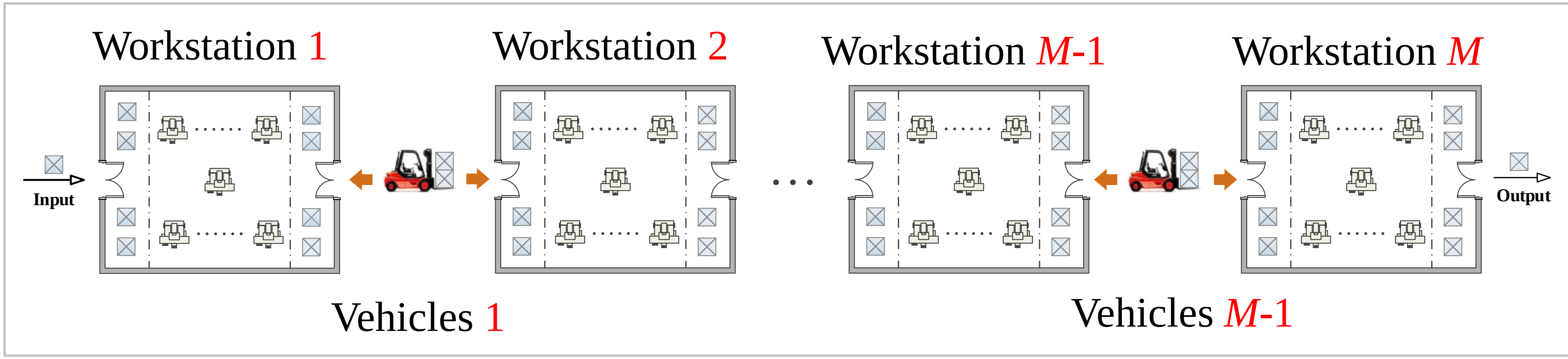
- Existing research on the design and analysis of production systems and material handling systems lacks an in-depth consideration of the **integration of these two areas**.

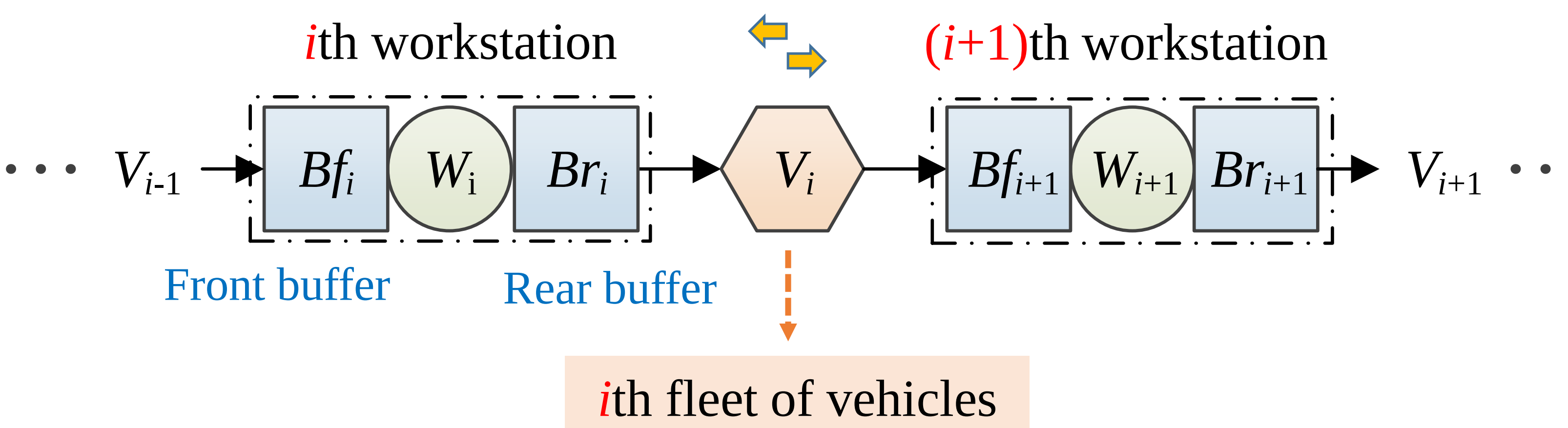
- Queueing (network) models:
  - Isolated queueing models with finite buffers and bulk arrival or service.
  - Closed queueing network models *without batch transfer*.
  - Open queueing network models with batch transfer and *infinite buffers* .

Open queueing network with finite buffers and **state-dependent batch transfer**

2 Model description

2.1 System description & assumptions





2 2.1 System description & assumptions

Service discipline: “First-come-first-served”

Loading discipline: “Either empty (buffer) or full (vehicle)”

State-dependent batch transfer: the transferred batch size depends on the current number of jobs at the buffers and the capacity of transporters.

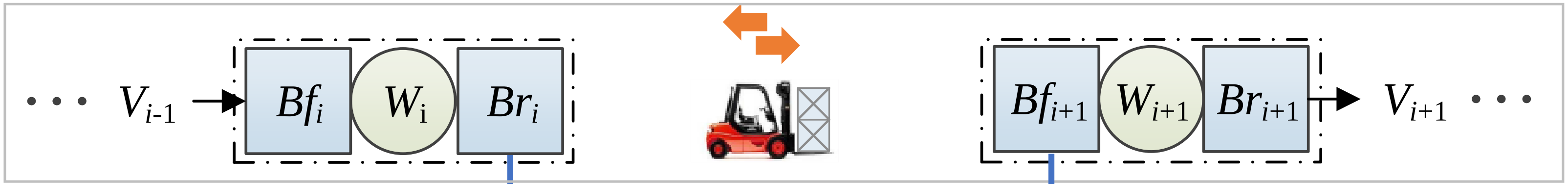
Unloading discipline: “Wait until empty (vehicle)”

- $r \geq x$ ➡ moving back
- $r < x$ ➡ waiting

r := the current number of vacancies at Bf_{i+1}.

x := the transferred batch size..

2.1 System description & assumptions



Jobs Blocking:	Production Blocking	Delivery Blocking
Vehicles Idling:	Pick-up Idling	Drop-off Idling

- External jobs arrival : Poission Process (λ).
- Processing times at each workstation : Exponentially distributed (μ_i).
- Pure travel times of vehicles (forward or backward) : Independent and exponentially distributed (τ_i).

Exponential transporting rate of vehicles : $v_i=(2\tau_i)^{-1}$.

2 Model description

2.2 Optimization problem formulation

Vehicle Configuration Problem

$$\boldsymbol{X}^* = arg\,min\ Q(\boldsymbol{X}) = arg\,min \sum_{i=1}^{M-1} cx_i,$$ (To minimize the total investment cost)

s.t.	$\Theta(\boldsymbol{X}) \geq \Theta_{\min},$	(The mean throughput constraint)
	$\mathrm{T}(\boldsymbol{X}) \leq \mathrm{T}_{\max}.$	(The mean sojourn time constraint)

x_i := The number of vehicles in the ith fleet, $x_i \in N^+$.

$\boldsymbol{X}$:= The vector of vehicle configuration, $\boldsymbol{X}=\{x_i\}$.

c := The unit price of a vehicle.

Q := Total investment cost.

Θ := System mean throughput.

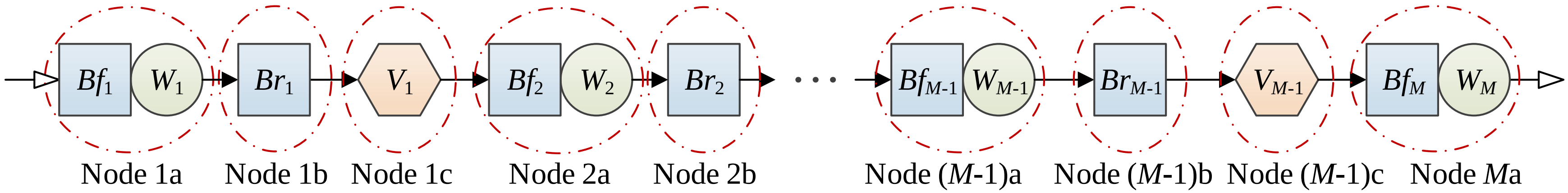
T := System mean sojourn time.

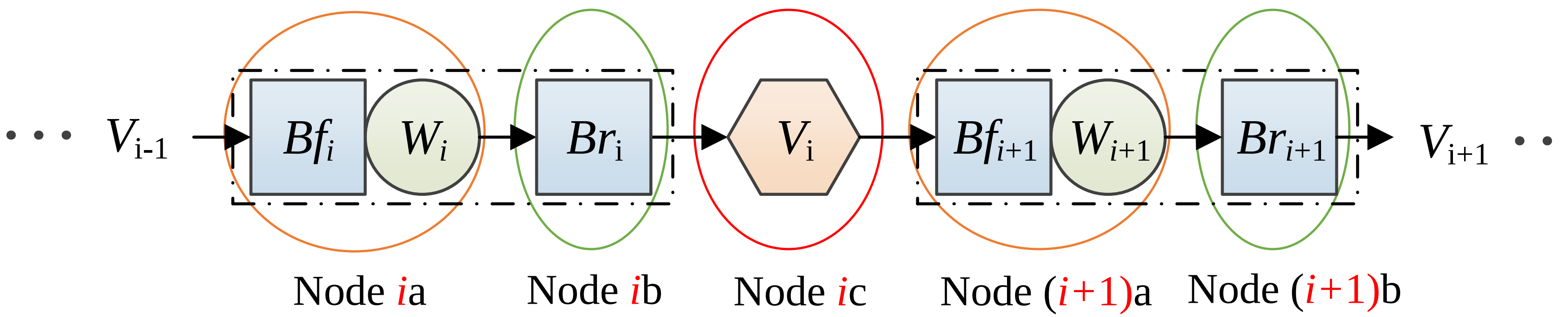
No closed-form functions are available for $\Theta(\boldsymbol{X})$ and $\mathrm{T}(\boldsymbol{X})$.

Decomposition of State Space Method

3 Proposed performance evaluation method

3.1 Queueing network model





Node *i*a ($i=2,3,\cdots,M$) : Batch arrivals with individual processing

Node 1a : Individual arrival with individual processing

Node *i*b ($i=1,2,\cdots,M$) : Individual arrival with batch processing

Node *i*c ($i=1,2,\cdots,M-1$) : Batch arrivals with batch processing

3.2 Formulation of the sub-systems

I. State Space

Status of vehicles

Status	Meaning	
1	Busy Trip	: Moving downstream toward the drop-off location.
1^-	Drop-off Idling	: Waiting at the drop-off location.
0	Empty Trip	: Moving upstream toward the pick-up location.
0^+	Pick-up Idling	: Waiting at the pick-up location.

Node *ic*

$S_{ic}\{(n_{ic}, s_i);$

$0 \le n_{ic} \le C_i;$ n_{ic}:= the number of jobs in the vehicle.

$s_i = 1, 1^-, 0, 0^+\}$ s_i := the status of the vehicle.

Node *ib*

$S_{ib}\{(n_{ib}, s_i);$

$0 \le n_{ib} \le Nr_i;$ n_{ic}:= the number of jobs at the rear buffer.

$s_i = 1, 0, 0^+\}$ s_i := the status of the related vehicle.

3.2 Formulation of the sub-systems

Node *ia*
($i \neq 1$)

$S_{ia}\{(n_{ia}, s_{i-1}, w_i);$

$0 \le n_{ia} \le Nf_i + 1,$ n_{ic}:= the number of jobs at the node.

$s_{i\text{-}1} = 1, 0,$ $s_{i\text{-}1}$:= the status of the related vehicle.

$w_i = 1, 0\}$ w_i := the condition of the server.

$S_{ia}\{(r_{i-1}, s_{i\text{-}1}, w_i);$

$0 \le r_{i-1} \le C_{i-1},$ $r_{i\text{-}1}$:= the number of remaining jobs in the vehicle.

$s_{i\text{-}1} = 1^{-},$ $s_{i\text{-}1}$:= the status of the related vehicle.

$w_i = 1, 0\}$ w_i := the condition of the server.

- Processes of transition between states: Exponential process
- **Continuous-Time Markov Chains** (CTMC**)** for each sub-system

- A fleet of X equivalent vehicles can be approximated by a single vehicle traveling X-times faster (Srinivasan et al. 1994).

3

3.2 Formulation of the sub-systems

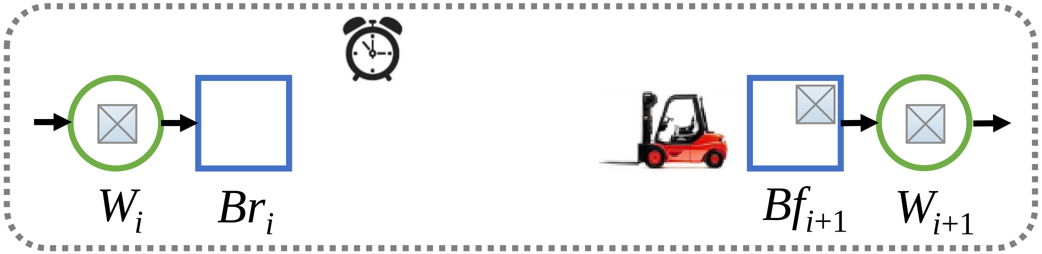
II. State Transition

n_{ib} := the current number of jobs at the buffer.

C_i := the maximum capacity of the vehicle.

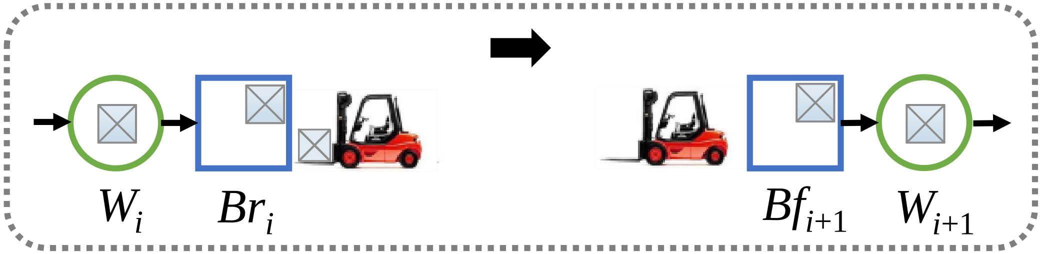
① State Transition: Empty Trip → *Pick-up Idling*

$n_{ib} = 0$

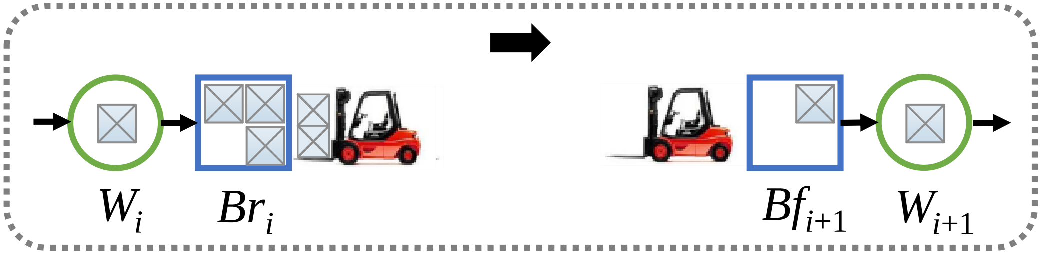


② State Transition: Empty Trip → *Busy Trip*

$n_{ib} < C_i$

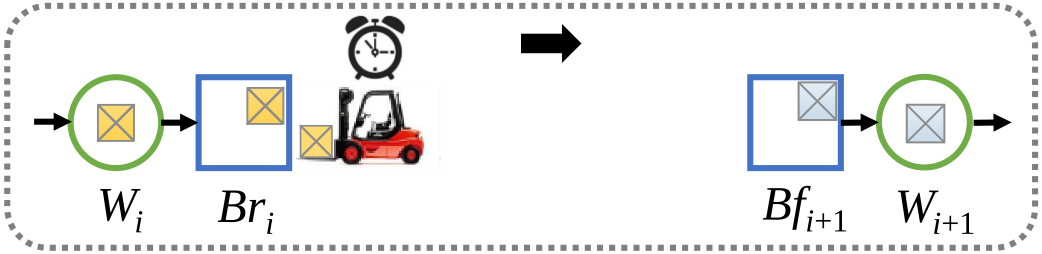


$n_{ib} \geq C_i$



II. State Transition

③ **State Transition:** **Pick-up Idling → *Busy Trip***

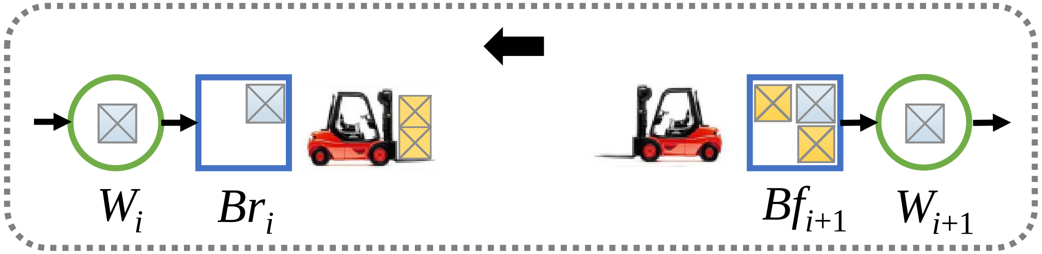


3.2 Formulation of the sub-systems

n_{ic} := the # of jobs that the vehicle carried. k := the # of vacancies at the buffer.

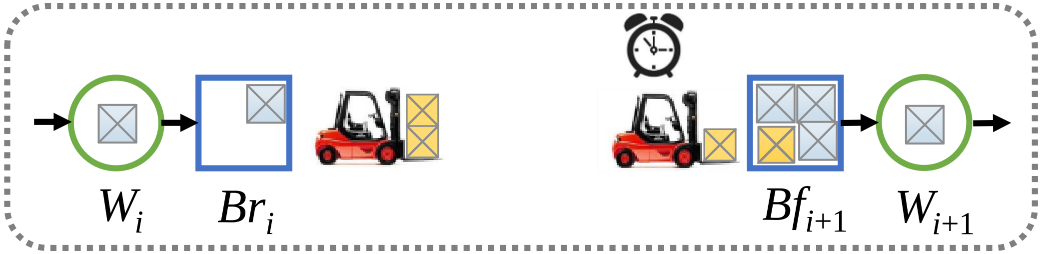
④ State Transition: Busy trip → *Empty Trip*

$n_{ic} \le k$

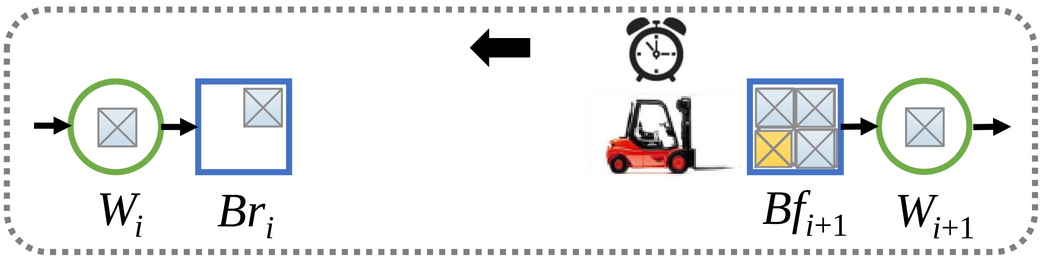


⑤ State Transition: Busy trip → *Drop-off idling*

$n_{ic} > k$



⑥ State Transition: Drop-off idling → *Empty Trip*



III. Rate Balance Equations

State Transition Diagram:

Node *ic*

$$2v_i \times \pi_{S_{ic}(0,0)} = \sum_{n_{ic}=1}^{C_i}\left\{2v_i\left[1-\sum_{k=1}^{n_{ic}} p_{(i+1)a}(k)\right]\times\pi_{S_{ic}(n_{ic},0)}\right\} + \mu^*_{i+1}\times\pi_{S_{ic}(1,1^-)}; \quad (1)$$

$$\lambda^*_{ib}\times\pi_{S_{ic}(0,0^+)} = 2v_i p_{ib}(0)\times\pi_{S_{ic}(0,0)}; \quad (2)$$

$$2v_i\times\pi_{S_{ic}(1,1)} = 2v_i p_{i(b,c)}(1)\times\pi_{S_{ic}(0,0)} + \lambda^*_{ib}\times\pi_{S_{ic}(0,0^+)}; \quad (3)$$

$$2v_i\times\pi_{S_{ic}(n_{ic},1)} = 2v_i p_{i(b,c)}(n_{ic})\times\pi_{S_{ic}(0,0)}, 2\le n_{ic}\le C_i; \quad (4)$$

$$\mu^*_{i+1}\times\pi_{S_{ic}(n_{ic},1^-)} = \sum_{z=n_{ic}}^{C_i}\left\{2v_i p_{(i+1)a}(z-n_{ic})\times\pi_{S_{ic}(n_{ic},1)}\right\} + \mu^*_{i+1}\times\pi_{S_{ic}(n_{ic}+1,1^-)}, 1\le n_{ic}<C_i; \quad (5)$$

$$\mu^*_{i+1}\times\pi_{S_{ic}(C_i,1^-)} = 2v_i p_{(i+1)a}(0)\times\pi_{S_{ic}(C_i,1)}. \quad (6)$$

$$\pi_j \mathbf{P}_j = \mathbf{0}$$

P := transition matrix of node *j*.

π_j := steady-state probabilities of node *j*.

3 Proposed performance evaluation method

3.3 Iterative algorithm for solution

Step 1. Initialization for each node

- Steady-state probabilities $\boldsymbol{\pi}_j^{(0)} = 0$.
- Probability parameters for other nodes = 0.

↓

Step 2. Iteration loops for each node

- State space $S_j\{\cdot\}$.
- Rate balance equations.
- Equivalent transition matrix $\mathbf{P}_j^{(r)}$.
- Solve $\boldsymbol{\pi}_j^{(r)}\mathbf{P}_j^{(r)}=\mathbf{0}$ using Gauss-Seidel method.
- Update the probability parameters for other nodes.

↓

$\boldsymbol{\Pi}^{(r)}=\{\boldsymbol{\pi}_j^{(r)}\}$

$norm\ \{\boldsymbol{\Pi}^{(r)}- \boldsymbol{\Pi}^{(r\text{-}1)}, inf\} >10^{-8}$?

N → $r=r+1$ (back to Step 2)

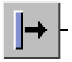
Y ↓

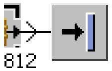
Step 3. System performance measures

- Steady-state probabilities of each node.

Proposed performance evaluation method

3.4 Computational experiments for performance analysis



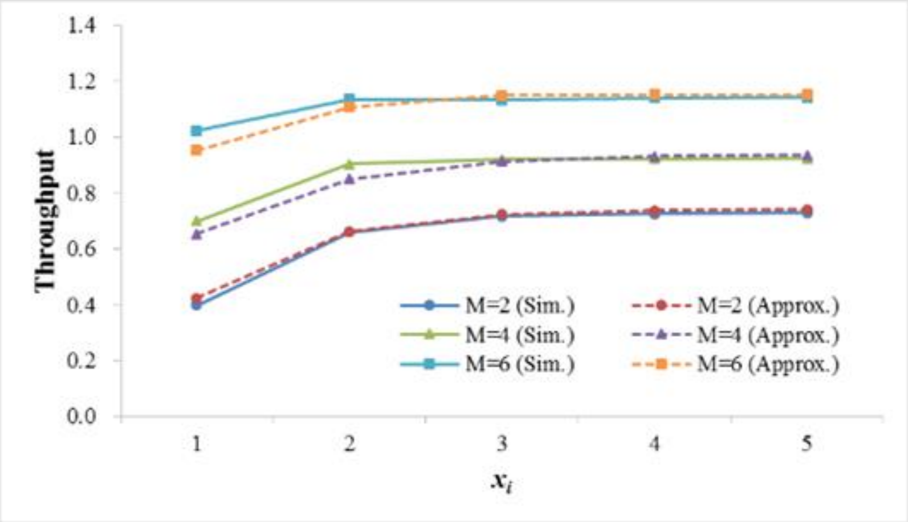


M	λ	μ_i	Nf_i= Nr_i	C_i	v_i	x_i	Throughput (Θ)			Sojourn time (T)		
							Sim.	Approx.	Δ%	Sim.	Approx.	Δ%
2	0.8	1.0	4	2	0.2	1	0.397	0.423	6.61	23.765	21.763	-8.42
						2	0.659	0.661	0.27	12.657	12.973	2.50
						3	0.718	0.724	0.83	9.918	10.589	6.76
						4	0.725	0.737	1.69	8.996	9.231	2.62
						5	0.729	0.741	1.58	8.664	8.182	-5.56
4	1.0	1.2	5	3	0.3	1	0.698	0.653	-6.50	36.575	33.829	-7.51
						2	0.902	0.849	-5.94	23.410	25.728	9.90
						3	0.919	0.913	-0.60	19.911	22.052	10.75
						4	0.920	0.931	1.13	18.306	19.651	7.35
						5	0.923	0.934	1.22	17.619	16.934	-3.89
6	1.2	1.5	6	4	0.4	1	1.023	0.953	-6.83	43.420	41.694	-3.98
						2	1.136	1.107	-2.54	28.766	32.031	11.35
						3	1.134	1.148	1.25	24.140	27.070	12.14
						4	1.138	1.151	1.10	22.492	24.312	8.09
						5	1.142	1.151	0.83	22.051	20.608	-6.54

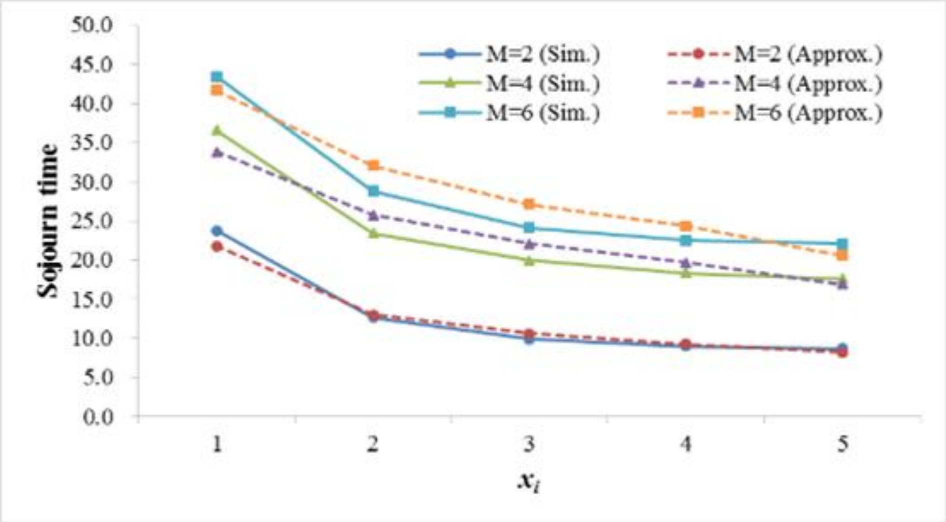
M := the number of workstations.

x_i := the number of vehicles in each fleet.

3 3.4 Computational experiments for performance analysis



(a) Throughput curves



(b) Sojourn time curves

	Throughput $\Theta(X)$	*Sojourn time* $T(X)$
$X\uparrow$	↑	↓

4 Optimization algorithm for vehicle configuration

4.1 Algorithm procedure

Vehicle Configuration Problem: **Monotone Integer Programming (MIP)**

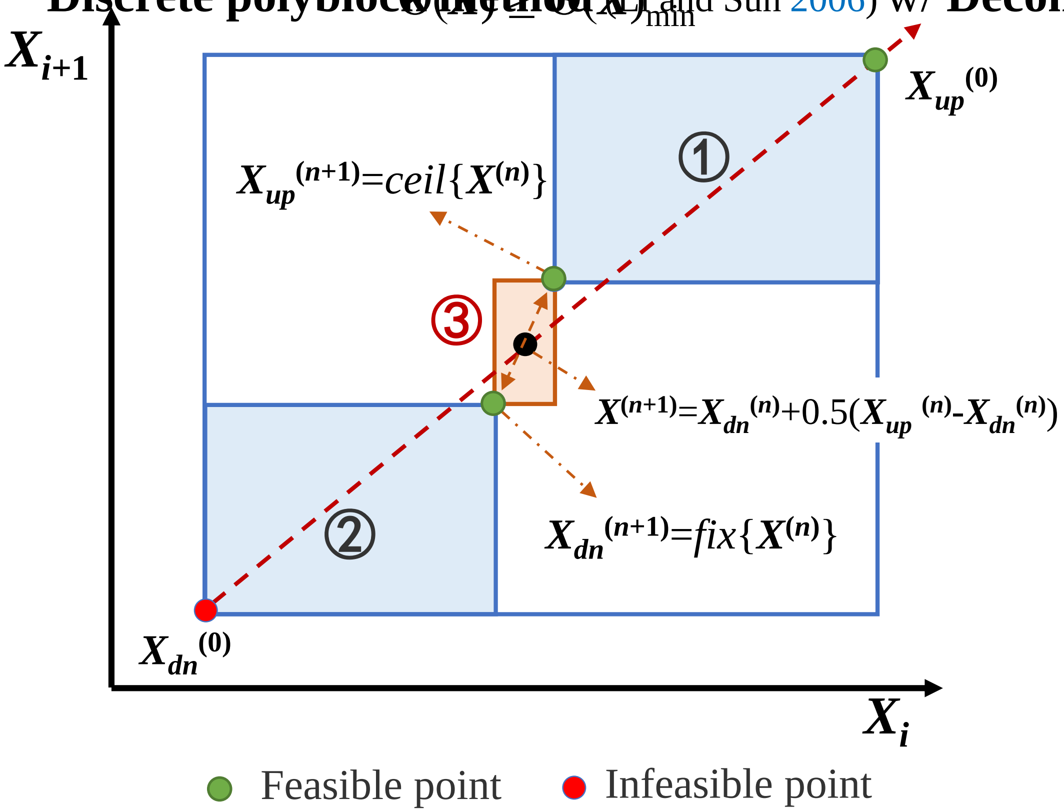
Step 1: Initialization

Step 2: Partitioned domain compression

Step 3: Enumerating-search

Discrete polyblock method (Li and Sun 2006) w/ **Decomposition of State Space Method**

$\Theta(\boldsymbol{X}) \geq \Theta(\boldsymbol{X})_{min}$



Integer Box for $\Theta(\boldsymbol{X})$: $\{(\boldsymbol{X}_{up})_\Theta, (\boldsymbol{X}_{dn})_\Theta\}$

↓

Integer Box for $T(\boldsymbol{X})$: $\{(\boldsymbol{X}_{up})_T, (\boldsymbol{X}_{dn})_T\}$

↓

$\boldsymbol{X}_{up}^{*}=max\{(\boldsymbol{X}_{up})_\Theta, (\boldsymbol{X}_{up})_T\}$

$\boldsymbol{X}_{dn}^{*}=max\{(\boldsymbol{X}_{dn})_\Theta, (\boldsymbol{X}_{dn})_T\}$

↓

$\boldsymbol{X}_{opt} \in \{\boldsymbol{X}_{up}^{*}, \boldsymbol{X}_{dn}^{*}\}$

4 Optimization algorithm for vehicle configuration

4.2 Computational experiments for vehicle configuration

M	X_{opt}	Throughput (Θ)			Sojourn time (T)		
		Sim.	Approx.	Δ%	Sim.	Approx.	Δ%
2	3	1.693	1.701	0.46	6.691	6.477	-3.20
3	4 3	1.690	1.695	0.32	11.289	12.199	8.06
4	3 4 3	1.689	1.663	-1.54	16.331	17.194	5.29
5	3 4 4 3	1.686	1.663	-1.34	20.750	21.933	5.70
6	3 4 3 4 3	1.688	1.662	-1.55	26.204	27.081	3.35
7	4 4 3 3 4 3	1.691	1.701	0.62	31.189	31.779	1.89
8	4 4 3 3 3 4 3	1.695	1.701	0.34	36.952	36.782	-0.46
9	4 4 3 3 3 3 4 3	1.694	1.701	0.42	40.251	41.754	3.73
10	4 4 3 3 3 3 3 4 3	1.696	1.701	0.30	44.265	46.708	5.52

External arrival rate: λ=1.8 units/hour, Vehicle capacity: C_i=3 units,
Processing rate: μ_i=2.1 units/hour, Vehicle transporting rate: v_i=0.35 units/hour,
Buffer capacity: Nf_i =Nr_i =7 units, Vehicle unit cost: c_i =$ 12,000.

5 Conclusions

- An integrated production and material handling serial system with state-dependent batch transfer is considered.
- A decomposition of state space method is proposed for system performance modeling.
- A vehicle configuration problem is solved by the discrete polyblock method embedded with the decomposition of state space method.
- Experiments show the accuracy and effectiveness of the proposed approaches.

Continuous-time Markov Chain

↓

Future works:

Generally distributed inter-arrival and service processes

Vehicle Configuration in an Integrated Material Handling System with State-dependent Batch Transfer

Thank you!

Hui-Yu ZHANG; Shao-Hui XI; Qing-Xin CHEN; Xiang LI; Ai-Lin YU

Guangdong University of Technology

Department of Industrial Engineering

Guangzhou, CHINA