

# Chapter 13

## Dual-Command Operation Generation in Bi-Directional Flow-Rack Automated Storage and Retrieval Systems with Random Storage Policy

Zhuxi Chen and Yun Li

**Abstract** In the bi-directional flow-rack (BFR) automated storage and retrieval systems (AS/RS), bins slope to opposite directions to make unit-loads be retrieved from half bins and be stored to the other half on the same working face. For random storage policy, a batching-greedy heuristic (BGH) has been proposed to generate dual-command (DC) operations in BFR AS/RS. In this paper, a novel DC operation generation rule specially designed for the BFR AS/RS is introduced to BGH, of which the effectiveness and efficiency are evaluated by simulation experiments.

**Keywords** Bi-directional flow-rack · AS/RS · Dual-command operation

### 13.1 Introduction

Automated storage and retrieval systems (AS/RSs) have been widely arranged in unit-load warehouses [1, 2] to obtain high storage accuracy, high throughput capacity and reasonable labor cost. For improving the floor space utilization, double-deep AS/RS [3], flow-rack AS/RS [4], 3D compact AS/RS [5], mobile rack AS/RS [6] and puzzle-based AS/RS [7] have been designed since 2005.

The bi-directional flow-rack (BFR) AS/RS was modified from the flow-rack AS/RS [8]. In a BFR, bins slope to opposite directions to ensure that unit-loads can be retrieved from and stored to both working faces. Therefore, dual-command (DC) operation, which stores a unit-load and retrieves another one within a single cycle to reduce the travel time of handling machine [9], can be generated in BFR AS/RSs. A batching-greedy heuristic (BGH) has been proposed for DC operation generation in BFR AS/RS with random storage policy [8]. However, the generation rule, NN (Nearest Neighbor) [10], SL (Shortest Leg) [10], and SDC (Shortest DC time) [11], applied by BGH for DC operations are designed for single-deep AS/RSs. Therefore,

---

Z. Chen (✉) · Y. Li

College of Information Engineering, Yangzhou University, Yangzhou, China  
e-mail: zxchen@yzu.edu.cn

a novel rule considered the structure of BFR AS/RS is introduced to BGH and its effectiveness and efficiency are evaluated by simulation experiments in this paper.

The rest part is as follows. In Sect. 13.2, BGH and a novel selection rule are introduced. The effectiveness and efficiency of the selection rule are evaluated in Sect. 13.3, followed by some conclusions in Sect. 13.4.

### 13.2 BGH and the Novel Generation Rule

An  $L \times H \times M$  BFR is illustrate in Fig. 13.1. The BFR contains  $L$  columns ( $L$  is even) and  $H$  rows of bins. Each bin consists of  $M$  segments to store at most  $M$  unit-load. Bins slope from  $F_B$  to  $F_A$  in Column 1, 3,  $\dots$ ,  $L - 1$  and from  $F_A$  to  $F_B$  in Column 2, 4,  $\dots$ ,  $L$ . Correspondingly, unit-loads slide from  $F_B$  to  $F_A$  in odd columns and from  $F_A$  to  $F_B$  in even columns driven by gravity. In each bin, unit-loads follow FIFO (first-in-first-out) rule, which implies that outgoing unit-loads are blocked by blocking unit-loads with certain probability. Blocking unit-loads must be removed before retrieving requested unit-loads. Removed blocking unit-loads are stored to bins on the same working face.

Two handling machines are arranged to  $F_A$  and  $F_B$ , respectively.  $t_h$  is the travel time between two horizontal adjacent bins and  $t_v$  is the travel time between two vertical adjacent bins. Let  $(x_i, y_i)$  represent bin  $i$  locates at Column  $x_i$  and Row  $y_i$  and  $(x_j, y_j)$  represent bin  $j$  locates at Column  $x_j$  and Row  $y_j$ . Let  $c_i = \max\{t_h x_i, t_v(y_i - 1)\}$  ( $c_j = \max\{t_h x_j, t_v(y_j - 1)\}$ ) be the travel time between  $i$  ( $j$ ) and the P/D station. Similarly,  $c_{ij} = \max\{|x_i - x_j|t_h, |y_i - y_j|t_v\}$  is the travel time from  $i$  ( $j$ ) to  $j$  ( $i$ ). The normalization and the shape factors of an  $L \times H \times M$  BFR are  $T = \max\{t_h L, t_v H\}$  and  $b = \min\{\frac{t_h L}{T}, \frac{t_v H}{T}\}$ , respectively.

In the random storage policy, any stored unit-load can be requested and an incoming unit-load can be stored to any available bin (a bin contains at least one idle

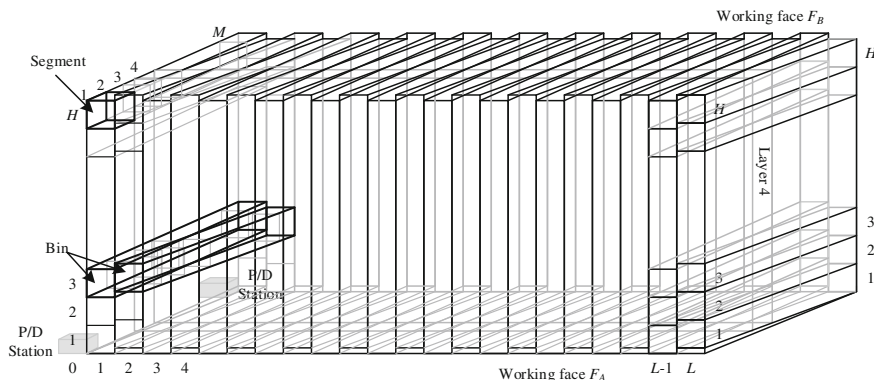


Fig. 13.1 An  $L \times H \times M$  BFR

segment). The sequencing of unit-loads is denoted as  $S = (s_1, s_2, \dots, s_N)$ , in which incoming unit-loads must be handled one by one. As well, the set of outgoing unit-loads is represented as  $R = \{r_1, r_2, \dots, r_N\}$ , in which outgoing unit-loads can be retrieved by any order.

The position of an idle segment is an opening and the position of an outgoing unit-load is a retrieval.  $I$  is the set of openings.  $J = \{j_1, j_2, \dots, j_N\}$  is the set of retrievals, in which  $r_k$  locates at  $j_k$  where  $k \in \{1, \dots, N\}$ . Let  $I_j$  be the set of openings located on the same working face of  $j$ .  $C_j$  is the travel time of removing and re-storing blocking unit-loads for retrieval  $j$ .

In a BFR AS/RS, a unit-load retrieved or removed from  $F_A$  ( $F_B$ ) releases an occupied segment on  $F_B$  ( $F_A$ ). Therefore, some openings on  $F_A$  ( $F_B$ ) become available after retrieving or removing unit-loads from  $F_B$  ( $F_A$ ). If such an opening is selected, the process time may be increased because the handling machine may stop to wait for the releasing.

BGH separates incoming unit-loads into batches, for each of which a sequence of DC operations are generated. The released openings in a batch are employed in the following batches to ensure the continuous working of handling machines. Let  $P = \lceil \frac{N}{n} \rceil$  be the number of batches. In the  $p$ th batch, a total of  $n$  or  $N - (P - 1)n$  incoming unit-loads and the same number of outgoing unit-loads are conducted by DC operations. BGH is described in Algorithm 1.

---

### Algorithm 1 Batching-greedy Heuristic

---

**Input:**  $S, R, I, J$

**Output:** sequence of DC operations

```

1: for ( $p = 1$  to  $P$ ) do
2:   Build  $S_p$ ;
3:    $S \leftarrow S - S_p, I_p \leftarrow \emptyset$ .
4:   while ( $S_p \neq \emptyset$ ) do
5:      $s \leftarrow \text{Head}(S_p), S_p \leftarrow S_p - \{s\}$ .
6:     Select  $j \in J$  and  $i \in I_j$  with minimal  $D_{ij}$ .
7:      $I \leftarrow I - \{i\}$ .
8:     while (blocking unit-loads exist) do
9:       Select  $i \in I_j$  with minimal  $c_{ij}$ .
10:       $I \leftarrow I - \{i\}, I_p \leftarrow I_p \cup \{j\}$ .
11:       $R \leftarrow R - \{r\}, J \leftarrow J - \{j\}, I_p \leftarrow I_p \cup \{j\}$ .
12:       $I \leftarrow I \cup I_p$ .
13: return
```

---

In Step 6 of BGH, an opening and a retrieval are selected based on the value of  $D_{ij}$ . In NN, SL and SDC,  $D_{ij} = c_{ij}$ ,  $D_{ij} = c_i + c_{ij}$  and  $D_{ij} = c_i + c_{ij} + c_j$ , respectively. The time complexity of Step 6 is  $O(LHN)$  for NN, SL, and SDC. Let  $\rho = \frac{N_u}{LHM}$  be the load rate, in which  $N_u$  is the number of stored unit-loads. For a bin, there are a number of  $K$  stored unit-loads. Because of the random storage policy, the expected value of  $K$  is  $E(K) = \rho M$ . Therefore, the expected number of blocking unit-loads for a retrieval is  $\sum_{k=0}^{K-1} \frac{1}{K} = \frac{\rho M - 1}{2}$ . The time complexity of choosing openings for blocking unit-loads of a retrieval is  $O(LH \frac{\rho M - 1}{2})$ . In normal,  $\frac{\rho M - 1}{2} < N$ , which makes the time complexity of generating a DC operation in BGH with NN, SL, and SDC is  $O(LHN)$ . Because

there are a total of  $N$  DC operations to be generated, the time complexity of BGH with NN, SL, and SDC is  $O(LHN^2)$ .

Because blocking unit-loads are re-stored in DC operations, SDCB (Shortest DC time with Blocking unit-loads) is designed, in which  $D_{ij} = c_i + c_{ij} + C_j + c_j$ . The time complexity of Step 6 of BGH with SDCB is  $O(\frac{\rho^{M+1}}{2}LHN)$ . Correspondingly, the time complexity of BGH with SDCB is  $O(\frac{\rho^{M+1}}{2}LHN^2)$ .

### 13.3 Performance Evaluation

In this section, the effectiveness and efficiency of SDCB are evaluated by simulation experiments. Because SL is better than NN and SDC [8], BGH with SL is executed in simulation experiments for comparing. Simulation experiments are coded in C++. A PC with 2.5 GHz CPU and 16 GB RAM is applied for running simulation experiments.

Let  $t_h = 1.0$  and  $t_v = 2.0$ . In each experiment, a total of 100 instances are conducted. For each instance, let  $N = 100$ , i.e., 100 DC operations are generated to store 100 incoming unit-loads and retrieve 100 outgoing unit-loads. The batch size takes value of  $\{2, 4, 10, 20\}$ . The DC operation travel times of instances are recorded, of which the average value is  $\bar{Z}_{DC}$ . As well, the process times of instances are saved, of which the average value is  $\bar{Z}_{PT}$ . In order to narrow the tables, the values of  $\frac{\bar{Z}_{DC}}{N}$  and  $\frac{\bar{Z}_{PT}}{N}$  are listed.

Firstly, the impacts of  $b$  are analyzed, in which  $L \times H \approx 200$ ,  $M = 5$ , and  $\rho = 0.8$ . The values of  $\frac{\bar{Z}_{DC}}{N}$  and  $\frac{\bar{Z}_{PT}}{N}$  obtained by BGH with SL and SDCB with different values of  $b$  are illustrated in Table 13.1.

**Table 13.1** Values of  $\frac{\bar{Z}_{DC}}{N}$  and  $\frac{\bar{Z}_{PT}}{N}$  obtained by SL and SDCB with different values of  $b$  ( $L \times H \approx 200$ ,  $M = 5$  and  $\rho = 0.8$ )

$b$		SL				SDCB				CPU time				
		2	4	10	20	2	4	10	20	2	4	10	20	
1.00	$\frac{\bar{Z}_{DC}}{N}$	30.33	30.43	30.77	31.25	30.71	30.82	30.97	31.20	SL	0.01	0.01	0.01	0.01
	$\frac{\bar{Z}_{PT}}{N}$	26.02	23.45	19.79	18.51	24.50	22.14	19.53	18.41	SDCB	2.14	2.12	2.10	1.87
0.67	$\frac{\bar{Z}_{DC}}{N}$	31.72	31.82	32.22	32.72	32.05	32.12	32.32	32.48	SL	0.01	0.01	0.01	0.01
	$\frac{\bar{Z}_{PT}}{N}$	27.71	24.70	20.71	19.49	25.71	23.19	20.40	19.15	SDCB	1.85	1.77	1.75	1.66
0.38	$\frac{\bar{Z}_{DC}}{N}$	38.09	38.24	38.80	39.35	38.27	38.34	38.55	38.89	SL	0.01	0.01	0.01	0.01
	$\frac{\bar{Z}_{PT}}{N}$	32.64	28.50	24.54	23.42	30.30	27.12	24.34	22.89	SDCB	1.81	1.78	1.64	1.58
0.25	$\frac{\bar{Z}_{DC}}{N}$	45.53	45.75	46.31	46.96	45.85	45.92	46.16	46.53	SL	0.01	0.01	0.01	0.01
	$\frac{\bar{Z}_{PT}}{N}$	38.22	33.09	29.08	27.70	36.03	32.22	28.81	27.51	SDCB	1.97	1.95	1.85	1.82

Table 13.1 indicates that the values of  $\frac{z_{DC}}{N}$  increase and the values of  $\frac{z_{PT}}{N}$  decrease with the increase of batch size. The values of  $\frac{z_{DC}}{N}$  obtained by SDCB become smaller than those gained by SL in large size of batch. The values of  $\frac{z_{PT}}{N}$  obtained by SDCB are smaller than those gained by SL. SDCB demands much longer CPU time than that spent by SL, which follows the time complexity. With the increase of  $n$ , the CPU time required by SDCB decreases.

Then, the impacts of different values of layers are examined and the experimental results are illustrated in Tables 13.2 and 13.3, respectively. In Table 13.2, let  $L \times H \times M \approx 1,000$ ,  $M \in \{10, 8, 6, 5\}$ ,  $b = 1.0$ , and  $\rho = 0.8$ . In Table 13.3, let  $L = 20$ ,  $H = 10$ ,  $\rho = 0.8$ , and  $M \in \{5, 6, 7, 8\}$ .

In Table 13.2, the values of  $n$ ,  $\frac{z_{DC}}{N}$  and  $\frac{z_{PT}}{N}$  have the similar changes as those in Table 13.1. SL returns shorter travel time than that gained by SDCB with small batch size and obtains higher travel time than that gained by SDCB with large batch size.

**Table 13.2** Values of  $\frac{z_{DC}}{N}$  and  $\frac{z_{PT}}{N}$  obtained by SL and SDCB with different values of  $M$  ( $L \times H \times M \approx 1,000$ ,  $b = 1.0$ , and  $\rho = 0.8$ )

M		SL				SDCB				CPU time				
		2	4	10	20	2	4	10	20		2	4	10	20
10	$\frac{z_{DC}}{N}$	25.78	25.91	26.53	27.89	26.59	26.85	27.49	28.69	SL	0.01	0.01	0.01	0.01
	$\frac{z_{PT}}{N}$	22.96	20.92	18.08	16.77	21.65	19.65	17.87	17.50	SDCB	1.36	1.24	1.20	1.17
8	$\frac{z_{DC}}{N}$	27.49	27.63	28.16	29.22	28.19	28.41	28.86	29.64	SL	0.01	0.01	0.01	0.01
	$\frac{z_{PT}}{N}$	24.12	21.93	19.00	17.52	22.79	20.72	18.46	17.86	SDCB	1.72	1.60	1.59	1.51
6	$\frac{z_{DC}}{N}$	28.39	28.50	28.86	29.46	28.90	29.01	29.27	29.65	SL	0.01	0.01	0.01	0.01
	$\frac{z_{PT}}{N}$	24.55	22.23	19.01	17.57	23.12	20.98	18.68	17.52	SDCB	1.85	1.76	1.75	1.53
5	$\frac{z_{DC}}{N}$	30.33	30.43	30.77	31.25	30.71	30.82	30.97	31.20	SL	0.01	0.01	0.01	0.01
	$\frac{z_{PT}}{N}$	26.02	23.45	19.79	18.51	24.50	22.14	19.53	18.41	SDCB	2.14	2.12	2.10	1.87

**Table 13.3** Values of  $\frac{z_{DC}}{N}$  and  $\frac{z_{PT}}{N}$  obtained by SL and SDCB with different values of  $M$  ( $L = 20$ ,  $H = 10$ , and  $\rho = 0.8$ )

M		SL				SDCB				Run time				
		2	4	10	20	2	4	10	20		2	4	10	20
5	$\frac{z_{DC}}{N}$	30.33	30.43	30.77	31.25	30.71	30.82	30.97	31.20	SL	0.01	0.01	0.01	0.01
	$\frac{z_{PT}}{N}$	26.02	23.45	19.79	18.51	24.50	22.14	19.53	18.41	SDCB	2.14	2.12	2.10	1.87
6	$\frac{z_{DC}}{N}$	31.29	31.43	31.81	32.35	31.69	31.79	31.96	32.31	SL	0.02	0.02	0.02	0.02
	$\frac{z_{PT}}{N}$	26.89	24.06	20.51	19.14	25.04	22.68	20.20	18.99	SDCB	3.00	2.90	2.84	2.72
7	$\frac{z_{DC}}{N}$	32.23	32.38	32.84	33.37	32.80	32.89	33.15	33.40	SL	0.03	0.03	0.03	0.03
	$\frac{z_{PT}}{N}$	27.56	24.96	21.08	20.07	26.11	23.56	20.98	19.66	SDCB	3.71	3.65	3.56	3.40
8	$\frac{z_{DC}}{N}$	33.41	33.56	34.11	34.82	33.87	34.03	34.29	34.68	SL	0.04	0.04	0.04	0.04
	$\frac{z_{PT}}{N}$	28.89	26.00	22.30	21.05	27.02	24.35	21.87	20.76	SDCB	4.77	4.70	4.62	4.49

**Table 13.4** Values of  $\frac{Z_{DC}}{N}$  and  $\frac{Z_{PT}}{N}$  obtained by SL and SDCB with different values of  $\rho$  ( $L = 20$ ,  $H = 10$ , and  $M = 10$ )

$\rho$		SL				SDCB				CPU time				
		2	4	10	20	2	4	10	20	2	4	10	20	
0.75	$\frac{Z_{DC}}{N}$	33.88	33.99	34.43	34.90	34.48	34.58	34.81	35.03	SL	0.03	0.03	0.03	0.03
	$\frac{Z_{PT}}{N}$	29.55	26.34	22.45	21.04	27.61	24.97	22.33	20.98	SDCB	5.65	5.58	5.46	5.44
0.80	$\frac{Z_{DC}}{N}$	35.10	35.27	35.94	36.74	35.71	35.83	36.16	36.69	SL	0.03	0.03	0.03	0.03
	$\frac{Z_{PT}}{N}$	30.40	27.36	23.30	22.06	28.42	25.57	22.81	21.81	SDCB	3.29	3.28	3.22	3.15
0.85	$\frac{Z_{DC}}{N}$	37.51	37.76	38.83	40.44	38.04	38.37	39.07	40.42	SL	0.02	0.02	0.02	0.02
	$\frac{Z_{PT}}{N}$	32.59	29.33	25.02	24.07	30.24	27.65	25.00	24.04	SDCB	5.22	5.16	5.01	4.77
0.90	$\frac{Z_{DC}}{N}$	41.46	41.97	43.77	46.48	42.03	42.66	44.79	47.89	SL	0.01	0.01	0.01	0.01
	$\frac{Z_{PT}}{N}$	36.33	32.58	28.82	28.04	33.59	30.63	28.86	29.12	SDCB	3.26	3.20	3.05	2.79

SDCB can obtain shorter process time than that gained by SL. As well, SL runs faster than SDCB. The CPU time demanded by SDCB decreases with the increase of  $n$ .

Table 13.3 demonstrates that the travel time increases and the process time decreases with the increase of  $n$ . SL obtains shorter travel time than that gained by SDCB. SDCB gains shorter process time that obtained by SL. SDCB requires much longer CPU time that demanded by SL. More layers in the BFR implies more blocking unit-loads, which causes the increase of operational cost and the increase of process time.

At last, the impacts of different values of  $\rho$  are evaluated by simulation experiments, in which let  $L = 20$ ,  $H = 10$ ,  $M = 5$ , and  $\rho \in \{0.75, 0.80, 0.85, 0.90\}$ . The experimental results are listed in Table 13.4.

Table 13.4 shows that the operational cost increases and the process time decreases with the increase of  $n$ . SL obtains lower operational cost than that gained by SDCB. SDCB gains shorter process time that obtained by SL. SDCB requires much longer CPU time that demanded by SL. Larger value of  $\rho$  implies more blocking unit-loads. Correspondingly, the values of  $\frac{Z_{DC}}{N}$  and  $\frac{Z_{PT}}{N}$  obtained by SL and SDCB increase with the increase of  $\rho$ .

In summary, the size of batch must be carefully evaluated to balance the operational cost and the process time of DC operations. SL obtains lower travel time and SDCB gains shorter process time. The process time may be more important than the travel time because the process time determines the response time, which is the key parameter of the quality of service (QoS) of a BFR AS/RS.

## 13.4 Conclusions

In this paper, the SDCB is introduced to the BGH for DC operation generation in BFR AS/RS with random storage policy. Experimental results illustrate that SDCB obtains shorter process time that gained by SL and SL gains shorter total travel time than that obtained by SDCB. The size of batch should be carefully considered to balance the operational cost and the process time. BGH with SDCB is more suitable than BGH with SL in applications where to improve the response time is more important than to reduce the operational cost.

In the future, dual-shuttle machine, which carries two unit-loads simultaneously and has been deployed to flow-rack AS/RSs for improving the retrieval performance [12], can be applied in BFR AS/RSs to enhance the throughput capacity.

**Acknowledgments** This research was supported by the Chinese National Natural Science Foundation (61402396), Natural Science Foundation of Jiangsu Higher Education Institutions (13KJB520027, 13KJB510039), The High-level Talent Project of “Six Talent Peaks” of Jiangsu Province (2012-WLW-024), Joint Innovation Fund Project of Industry, Education and Research of Jiangsu Province (BY2013063-10) and The Talent Project of Green Yangzhou & Golden Phoenix (2013-50).

## References

1. Van Der Berg JP (1999) A literature survey on planning and control of warehousing systems. *IIE Trans* 31(8):751–762
2. Roodbergen KJ, Vis IFA (2009) A survey of literature on automated storage and retrieval systems. *Eur J Oper Res* 194(2):343–362
3. Lerher T, Sraml M, Potrc I, Tollazzi T (2010) Travel time models for double-deep automated storage and retrieval systems. *Int J Prod Res* 48(11):3151–3172
4. Sari Z, Saygin C, Ghoulali N (2005) Travel-time models for flow-rack automated storage and retrieval systems. *Int J Adv Manuf Technol* 25(9):979–987
5. De Koster MBM, Le-Duc T, Yu Y (2008) Optimal storage rack design for a 3-dimensional compact AS/RS. *Int J Prod Res* 46(6):1495–1514
6. Chang TH, Fu HP, Hu KY (2006) Innovative application of an intergrated multi-level conveying device to a mobile storage system. *Int J Adv Manuf Technol* 29(9–10):962–968
7. Uludag O (2014) GridPick: a high density puzzle based prder picking system with decentralized control. Ph.D. Thesis, Auburn University
8. Chen Z, Li X, JND Gupta (2015) A bi-directional flow-rack automated storage and retrieval system for unit-load warehouses. *Int J Prod Res* 53(14):4176–4188
9. Gagliardi JP, Renaud J, Ruiz A (2014) On sequencing policies for unit-load automated storage and retrieval systems. *Int J Prod Res* 52(4):1090–1099
10. Han MH, McGinnis LF, Shieh JS, White JA (1987) On sequencing retrievals in an automated storage/retrieval system. *IIE Trans* 19(1):56–66
11. Lee HF, Samaniha SK (1996) Retrieval sequencing for unit-load automated storage and retrieval systems with multiple openings. *Int J Prod Res* 34(10):2943–2962
12. Chen Z, Xiaoping L (2015) Sequencing method for dual-shuttle flow-rack automated storage and retrieval systems. *J Southeast Univ (English Edition)* 31:31–37