TAAI 2024

Optimal Assignment of Immediate Tasks in Multi-Agent Pickup and Delivery

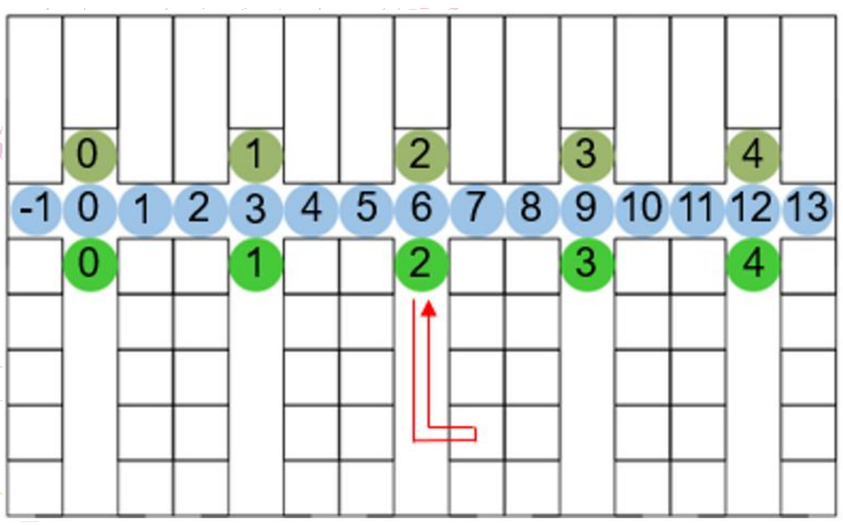
Taisei Hirayama[1], Itsuki Noda[1], Hiroki Sakaji[1], Norihiko Kato[2]
1) Hokkaido University, 2) Toyota Industries Corporation

Time: Dec. 06, 13:00 – 15:10
Room: Hall
Paper ID: 23

Objection in Automated Warehouses

In practical applications: **Need to accept immediate and emergent tasks and insert them proper position in planned queue**

Graph Representation of Warehouse



Delivery location
Entrance node (e_k)
Pickup location
Unit node (u_k)
Pathway
Horizontal aisle node

$k = 0,1,2,\cdots$

Agents deliver a payload, avoiding collisions with other agents

How to Interrupt?

General Method (Immediate First Method; IFM)

Insert immediate task on top of regular queue (TSK) of the delivery location and process it promptly

IFM's problems:
- Only considers immediate tasks
- Ignores delays of regular tasks

Optimal interruption methods improve overall efficiency

Introducing Procrastination

Proposed methods explore the optimal degree of procrastination from 0 to the $Procrastination\ Limit\ (PL)$ by tentative reassignment

Optimal Timing Search (OTS)

Insert immediate task on top of the TSK of the delivery location and process it at **optimal timing**

Optimal Vertical Location Search (OVLS)

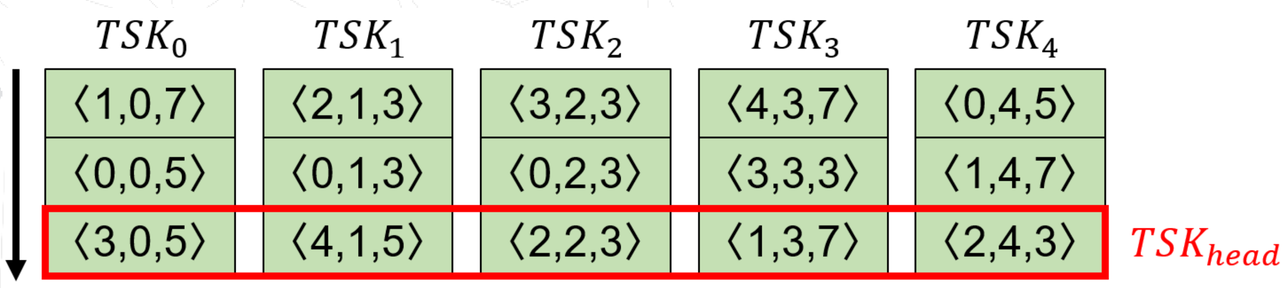
Insert immediate task to **optimal location** of the TSK of the delivery location

Optimal Horizontal Location Search (OHLS)

Insert immediate task on top of **optimal TSK**

Variance Minimization (VM)

Insert immediate task on top of the TSK with the fewest remaining tasks



- Each task is defined by a tuple $\langle u_i, e_j, t_{pickup} \rangle$
- Before the operation, all regular tasks are arranged in each task queue TSK_k of the delivery location e_k
- t_{pickup} is the time the unit node is occupied (red arrow in top figure)
- Agents select tasks from TSK_{head}

How to Find Optimum?

time window

Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Agent0	E 3	H 9	U 3	U 3	U 3	U 3	U 3	H 9	H 8	H 7	H 6	H 5	H 4	H 3	H 2	H 1	H 0	E 0	H 0	H -1	H -1	H -1
Agent1	E 2	H 6	U 2	U 2	U 2	H 6	E 2	H 6	H 5	H 4	H 3	H 2	H 1	H 0	U 0	U 0	U 0	U 0	U 0	H 0	E 0	E 0
Agent2	E 4	H 12	U 4	U 4	U 4	U 4	U 4	H 12	H 11	H 10	H 9	H 8	H 7	H 6	H 5	H 4	H 3	E 1	H 3	H 2	H 1	H 0

1 2 3 4

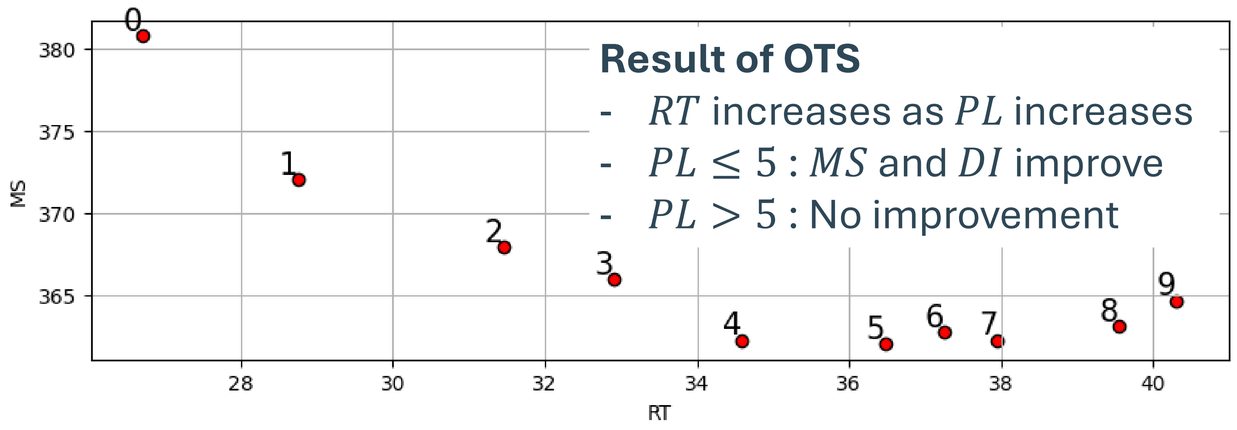
- When an immediate task occurs, regular tasks and immediate tasks are reassigned to agents
- The more tasks are reserved within the time window at the reassigning, the more optimal it is (E.g., four tasks are reserved ↑)

Evaluation by Simulation

Metrics (smaller values are preferred for all)

Makespan (MS) : Total time required to complete all tasks
Response Time (RT) : Average time from the occurrence of an immediate task to its completion
Delay per Immediate (DI) : Average delay time of individual tasks per immediate task

Experimental Results



Result of OTS
- RT increases as PL increases
- $PL \leq 5$: MS and DI improve
- $PL > 5$: No improvement

	Optimal Timing Search				*Optimal Vertical Location Search*		
PL	MS	RT	DI		MS	RT	DI
0	380.79	**26.71**	12.49	IFM	374.68	**35.62**	11.13
1	372.02	28.75	11.03		365.78	42.60	9.05
2	367.94	31.46	10.45		363.32	51.15	8.16
3	365.93	32.90	10.04		363.19	59.78	7.59
4	362.24	34.59	9.12		**362.89**	63.51	7.32
5	**362.00**	36.48	**8.96**		363.22	65.49	7.25
6	362.75	37.25	9.08		363.00	65.30	**7.23**
7	362.24	37.95	9.07		363.00	65.30	7.23

- $PL = 0$: In OTS, the algorithm is identical to IFM
- $PL > 5$: No improvement may be caused by simultaneous handling of multiple immediate tasks

	MS	RT	DI
Optimal Horizontal Location Search	357.41	32.93	8.00
Variance Minimization	369.47	40.80	9.25

- A trade-off between MS and RT is observed in other methods
- **MS and DI can generally be reduced by allowing some procrastination of immediate tasks**
- MS and DI have small values with OHLS and VM
- This indicates that **the impact of the constraint specifying the delivery location is significant**

TAAI 2024