A Simulation Model for Spatial Scheduling of Dynamic Block Assembly in Shipbuilding

Zhuo Liu[1], David K.H. Chua[2], and Keng Hwee Wee[3]

Abstract

Scheduling block assembly in shipbuilding poses great difficulties in accurate prediction of the required spatial resource and effective production control for achieving managerial objectives due to lack of optimal block configuration and the stochastic nature of production system. In this study, a look-ahead scheduling mechanism with an application of discrete-event simulation is proposed to solve the dynamic spatial scheduling problem. A systematic simulation-based framework is suggested for validating schedules generated at high-level planning together with a heuristic-based optimizer for improving spatial resource utilization. Through imitation of the dynamic spatial operation under various priority rules, statistical analysis of resultant performance can be conducted to select the best performing control policy. A case study with computational experiments is performed and results are reported as well.

Keywords: Discrete-Event Simulation, Block Assembly, Shipbuilding, Dispatching Rule, Spatial Scheduling

Introduction

Although shipbuilding industry shares similar characteristics with civil engineering and construction projects, most work-class ship producers have employed the principles of group technology (GT) to rationalize the dedicated production processes and produce similar interim products repetitively at the upstream (Gribskov et al., 1988; Spicknall, 1997). To maximally utilize the process commonality, the ship hull is divided into a few grand chunks, which are further segmented into a number of small pieces called blocks. Consequently, the entire shipbuilding is sequenced into on-site construction (e.g. erection) and series manufacturing (e.g. block assembly, outfitting) of blocks (Eyres, 2007).

The stage of block assembly, our focus of this study, involves a series of complicated processes (e.g. fitting, welding) on a bounded working area (Liker and Lamb, 2000). Figure 1 gives a snapshot of block assembly activities at shop floor. In practice, the assembly area, a critical spatial resource, is characterized by various types of jigs to support assembly jobs. However, the availability of locatable area for assembling one particular block depends on the dynamic spatial layout, block dimension and equipment constraints (e.g. crane) at shop floor. A large spectrum of researches has been conducted to investigate various algorithms for optimal configuration and develop decision support systems for spatial scheduling of dynamic block assembly. However, all the studies deal with the spatial scheduling problem in a deterministic environment, which hardly reflect the realistic production situation. Uncertainties in terms of the stochastic nature exist in almost each aspect of block assembly because of heterogeneous resource requirements,

---

[1] Postdoctoral Research Fellow, Department of Civil Engineering, National University of Singapore, Singapore, Tel.: +65-65166463, Email: cveliuz@nus.edu.sg

[2] Associate Professor, Department of Civil Engineering, National University of Singapore, Singapore, Tel.: +65-6516-2195 Email: cvedavid@nus.edu.sg

3 Senior vice president, Singapore Sembcorp Marine Ltd, Email: khwee@sembcorpmarine.com.sg

unpredictable disturbances and the labor-intensive nature of ship building. Thus, processing time of block assembly is a combining result of man-hour control, material handling, and owner qualification requirements. The generated plans together with spatial allocation schedule always lose their creditability as production proceeds. Although rescheduling provides a remedy means for reactively adjusting the production rhythm based on the real-time information, being blind to the executability of middle-term plan regarding the limited spatial constraints does not fundamentally solve the problem. Toward this, a look-ahead scheduling mechanism by considering uncertainties is desirable to predict the system performance based on the planned block list. Furthermore, various dispatching rules of assigning assembly jobs can be investigated to select the most suitable rule for the coming product mix and scheduling period.

Figure 1. Block assembly in workshop

In this paper, we propose a look-ahead spatial scheduling mechanism by modeling the stochastic events with discrete-event simulation model for effective risk hedging. The paper is organized as follows. After literature review on relevant research work, a systematic framework for look-ahead scheduling mechanism is presented, wherein a simulation model is developed and integrated with a heuristic-based algorithm for optimizing the spatial layout of block assemblies. A case study with computational experiment is then presented to demonstrate the proposed approaches.

Literature Review

Spatial Scheduling

Spatial scheduling exhibits certain similarity with two-dimensional packing problems in that the minimal waste is desirable while allocating a set of rectangular items to larger rectangular containers (Lodi et al., 2002). However, dynamic operational environment of spatial scheduling hinders the application of available methods dedicated for solving two-dimensional packing problems (Shin et al., 2008). As a result, various heuristic-based algorithms have been developed to optimize the block assignment and space allocation. Park et al. (1996) present a scheduling algorithm using partial enumeration and decomposition to generate a spatial allocation plan. Lee and Lee (1996) implement a spatial scheduling expert system for block assembly, wherein a methodology for spatial layout of polygonal objects is developed. Utilizing the similarity of the two-dimensional packing problem, Shin et al. (2008) present a bottom-left-fill heuristic method for spatial planning of block assemblies and the system has been implemented in Ryu et al. (2008).

Discrete-Event Simulation for Manufacturing Operation

A simulation modeling method provides the imitation of the operation of a real-world system overtime if the problem in question can not be solved analytically. In the past decades, a large body of simulation-based research has been made to address various issues related to manufacturing system design and operation. For instance, material handling system design has been an extremely popular research area for the application of simulation due to the complexity inherent in system component interaction and difficulty associated with analytical modeling method. Compared to long-term analysis of system design, operation planning and scheduling always involve a short-term decision within a dynamic and stochastic environment. The powerful capability of modeling the realistic production system makes simulation an attractive analysis and evaluation tools for on-line shop floor control. Son et al. (2002) present the integrated architecture for a simulation-based shop control system. Lee et al. (2007) propose a distributed top-floor control system to integrate and coordinate heterogonous simulation models. Lei et al. (2010) develop a Petri-Net based simulation package to support maintenance decision. The application of simulation model is also found in scheduling semiconductor wafer fabrication to predict system performance (Zhang et al., 2009).

Methodology

In this study, a look-ahead scheduling mechanism is introduced as an intermediate component between a long-term aggregate production planning (APP) and real-time shop floor control (SFC), which can be illustrated in Figure 2. By imitating shop floor operation of the plan generated from APP, the look-ahead module employs a spatial scheduling simulation model to select priority rule, predict the system performance and feedback spatial utilization to high-level planning. Simulation-based schemes are usually used in a rolling-horizon basis; that is, if new four week job release data are available and decision are made for 4 weeks using simulation, then only the first week decisions are implemented, and at the beginning of the next week new decisions are made for the following weeks using fresh real-time data. For the look-ahead scheduling, top-performing priority rule is decided for the coming scheduling period and product mix. Meanwhile, re-planning request will be sent to high-level planning if the planned jobs within the look-ahead horizon are found not executable regarding the spatial resource through simulation module.

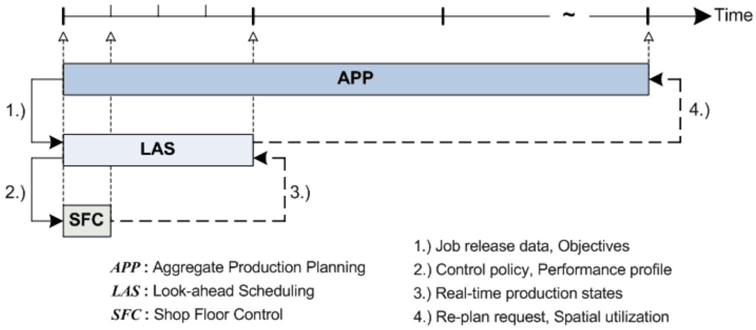

Figure 2. Schematic view of look-ahead scheduling in a hierarchical framework

Architecture of Simulation Model for Spatial Scheduling

The proposed look-ahead scheduling architecture is composed of a simulator, scheduler, optimizer, and databases containing information related to real-time system states, a set of dispatching rules and production plan generated from high-level planning stage.

Information included in each virtual database, as shown in Figure 3, may be physically stored in one database. *System States* store the real-time production information, e.g. the current positions of all block assemblies and their elapsed days. This information is updated as system states change. *Dispatching Rules* store a set of rules to determine the priorities of assembly work. *Production Plans* store high-level planning information about processing time, start date, due dates, required resource, and performance measure.

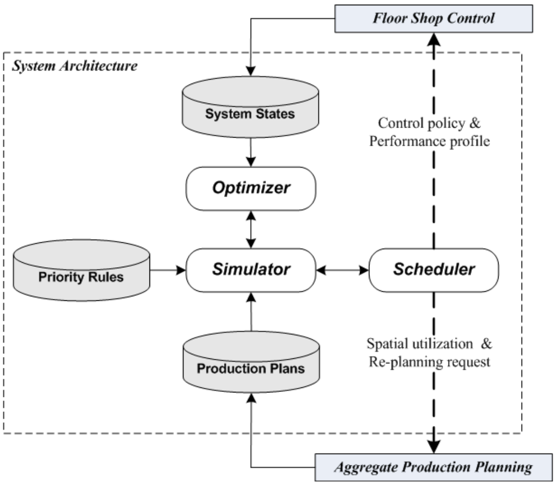

Figure 3. System Architecture of simulation-based spatial scheduling

The simulator, being parametric in nature, provides required flexibility to model the investigated production system. The simulator is requested by the scheduler to perform a series of discrete-event simulations with specified dispatching rules. During simulation, the simulator calls the spatial optimizer to seek an optimal location for block assembly being scheduled. Then the simulator sends to the scheduler simulation results of each running scenario where each dispatching rule is applied to the selected planning horizon.

Scheduler interprets the results and returns them to system users. A look-ahead view of spatial requirement for the coming workload, together with system performance, will be provided to further decide the optimal dispatching rule regarding certain criteria. If estimated system performance in each scenario is not satisfactory regarding pre-specified criteria, a feedback might be made to request re-planning. At the time the new assembly job arrives, the scheduler will call the spatial optimizer to assist in locating blocks.

Optimizer aims to improve area utilization of workshop and assists in locating blocks in working area when the job of one block assembly is released or one block assembly is completed. A heuristic-based algorithm is developed to search the optimal location for block assembly by considering positioning strategy and assembly requirements. Detailed algorithm to implement spatial scheduling is given at next section.

Optimizing Space Allocation for Block Assembly

In this study, an enumeration-based heuristic algorithm is developed for optimal block placement. Block and assembly area in shop floor are simply considered rectangle shapes. This assumption can be proven valid in practice that the components are always placed or assembled into larger sub-blocks around the irregular blocks so that an approximate rectangle area is bounded. As illustrated in the schematic assembly layout of Figure 4, assembly area is occupied by two blocks and it is very expensive to search all the points since the locatable space is continuous. However, corner points of objects and intersection points between object edge extensions are believed to provide a set of meaningful discrete points out of the continuous space (Lee and Lee, 1996), which effectively reduce the search complexity. Here, $C = \{c_1, c_2, c_3, c_4\}$ represents the set of the corner points of assembly area. $V^i = \{v_1^i, v_2^i, v_3^i, v_4^i \mid i=1, 2\ldots\}$ denotes the set of corner points of block *i*. $E^i = \{e_1^i, e_2^i \ldots \mid i=1, 2\ldots\}$ denotes the set of points of intersection between block or assembly area edges and block edge extensions. Hence, through the union of the above point sets, all the vertices of the empty space in the assembly area can be derived as

$$\xi = V^1 \cup \ldots \cup V^i \cup \ldots \cup E^1 \cup \ldots \cup E^i \cup \ldots \cup C\,. \tag{1}$$

Theoretically, the maximal size $|\xi^*|$ of ξ can be derived as follows:

$$|\xi^*| = 4n^2 + 4n + 4 \tag{2}$$

where n is the number of existing blocks on work area.

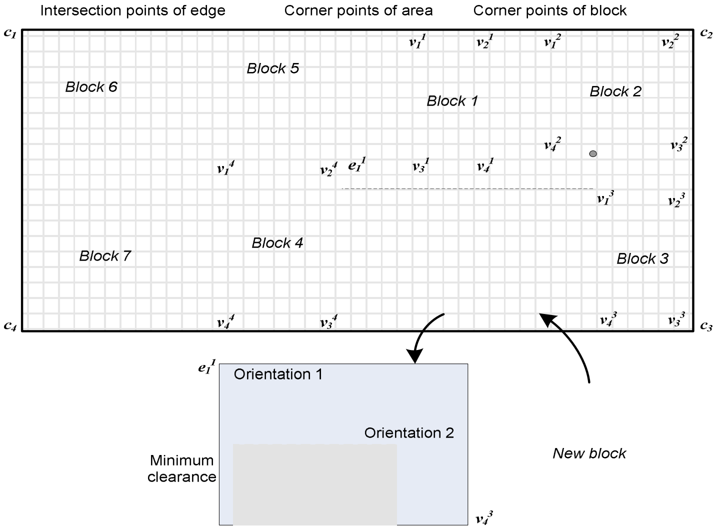

Figure 4. Illustration of spatial optimizer for positioning block assembly

By selecting any two points from the set of ξ as the diagonal vertices of rectangle space, a set of feasible space can be constructed, say *S*, while satisfying the requirements of non-interference with occupied area. In the meantime, a reduction procedure can be imposed to narrow down the range of feasible spaces according to the required area of block and minimum clearance of adjacent blocks. Consequently, search an optimal space for one block assembly can be converted into an enumeration problem which is to minimize the area redundancy between one feasible space and the required area for assembly.

After a locatable area represented by two diagonal vertices is identified for one block assembly, positioning strategies are used to decide the reference point and the orientation for the assembly job. To minimize the fractured space, edging strategy is adopted here to overlap one ore two of the block edges with working area edges (Lee & Lee, 1996) and then left-down-most strategy is used to decide the reference point (Shin et al., 2008). In addition, maximal remnant rectangle space utilization is employed to maximize the free rectangular space while determining the block orientation (Lee and Lee, 1996). As illustrated in Figure 4, for the new block assembly job and identified area represented by e_1^1 and v_4^3, orientation 1 is preferable since the remnant rectangle area is obviously larger than the one resulted from orientation 2.

Optimizing block placement is driven by two events: new job arrival and job completion and their workflows are depicted in Figure 5. A list of workshops satisfying the assembly requirements of the new job and a list of waiting jobs satisfying the capacities of released workshop are formulated for job arrival and job completion, respectively. In case of new job arrival, balancing loads among workshop is considered here to select a workshop with the lowest area utilization when a few working areas in different workshop are available. The area utilization $\mu(l,t)$ of workshop l at time t is defined as: $\mu(l,t)$ =(the sum of the areas of the blocks being assembled in workshop l at time t)/(area of the workshop l). On the other hand, when a block assembly is completed, a priority rule has to be used to select a waiting job from the list. Then all the vertices are enumerated according to the real-time block configuration in the selected workshop and the rectangle space with minimal waste is selected based on the required assembly area for the assembly job.

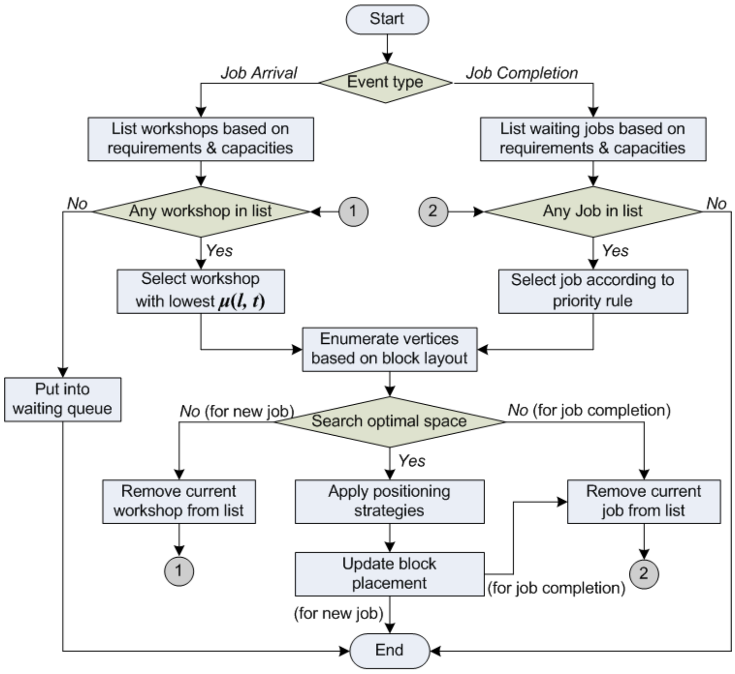

Figure 5. Flowchart of optimizing block placement for job arrival and completion

Case Study

To demonstrate the proposed approach, a case study involving a local shipyard to dynamically schedule the location of block assemblies is introduced. In the investigated assembly system, block assembly jobs are supposed to release according to high-level planning and however, realistically arrive randomly within a time range. Each job of assembly requires one certain working area in workshop depending on block size and required equipments. The required area is assumed to be known and deterministic. The occupied spatial resource will not be released until the job is completed. The processing time for each assembly has been estimated individually at high-level planning and however, is assumed to follow a discrete uniform distribution within a time range.

For the simulation experiments, the job release of block assembly is generated from a discrete uniform distribution characterized by DU(AST_i, Dev), where AST_i is the scheduled assembly start date and Dev is the range. That means block assembly is released following a discrete uniform distribution with a range from $AST_i+ Dev$ to $AST_i- Dev$. Processing time of block assembly is supposed to follow a discrete uniform distribution characterized by DU($DAYS_i$, PER), with a rage from $DAYS_i* (1-PER)$ to $DAYS_i*(1+PER)$. In this study, Dev and PER are estimated from the historical data in the shipyard and take 3 days and 10%, respectively. Twenty instances are randomly generated and system performance data (e.g. waiting time, tardiness, storage volume) are statistically collected.

From the extensive research of the priority dispatching rules, it is soundly concluded that there is no single rule that yields the best performance in all conditions and the performance of a rule highly depends on the managerial objectives and production status (Blackstone et al., 1982). Therefore, seeking a dispatching rule within certain period based on real-time system condition and the performance of interest can improve the overall performance for the entire planning horizon. In this study, a series of simulation runs is performed using each of the candidate priority rules, as shown in Table 1, at the time of executing the look-ahead scheduling. By comparing their performance measures, the rule that produce the best performance is selected and applied into floor shop control until the next scheduling is triggered.

Table 1. Dispatching priority rules (for assembly job *i* at time *t*)

Priority Rule	**Description**	**Definition**
FCFS	First Come First Served	$FCFS = \min\{r_i\}$
SFT	Shortest Processing Time	$SFT = \min\{p_i\}$
SLACK	The Least Slack	$SLACK = \min\{d_i - p_i - t\}$
EDD	Earliest Due Date	$EDD = \min\{d_i\}$
CR	Critical Ratio	$CR = \min\{(d_i - t)/p_i\}$
MWMT	Most Waiting time Multiplied by Ton	$MWMT(t) = \min\{(t - r_i) \times w_i\}$
MAR	Minimal Area Residue	$MAR(t) = \min\{A(t) - a_i\}$
RANDOM	Randomly Select a job	

Nomenclature:

p_i: Processing time of block assembly *i*
d_i: Due date of block assembly *i*
r_i: Arrival time of block assembly *i*
w_i: Weight (tons) of block *i*
a_i: Area of block assembly *i*
$A(t)$: Maximal locatable area at time *t*

40 assembly jobs within 5-week look-ahead horizon have been collected from a local shipyard, as shown in Appendix. Obviously, these jobs exhibit heterogeneous requirements in assembly area, execution date and processing time. The performance considered in this study includes total tardiness (TT), total storage (TS) in the queue and mean area utilization (MAU). There performance measures are as follows.

$$TT = \sum_{i=1}^{N} \max(c_i - d_i, 0),\ \ TS = \sum_{j=1}^{L}\sum_{i=1}^{N} w_i q_{ij},\ \ MAU = \left[\sum_{l=1}^{L}\sum_{j=1}^{P}\sum_{i=1}^{N}\left(a_i b_{ijl}\right)\right] \Big/ L \cdot \sum_{l=1}^{P} S_l \tag{3}$$

where c_i is the actual completion time; N is the number of assembly jobs under consideration. w_i is the weight of block i; q_{ij} is the binary state variable to indicate whether block i in the waiting queue at day j; L is the number of days within the look-ahead horizon. a_i is the required assembly area for block i; b_{ijl} is the binary state variable to indicate whether block i is assembled at workshop l at day j; S_l is the work area at workshop l.

For each of the 8 priority rules, the simulation outputs for the twenty replications are averaged, as presented in Table 2. As to total tardiness, it is obviously found that the employment of SLACK rules produces the smallest mean value as well as standard deviation. In addition, MWMT, FCFS and CR generate very close performance with total tardiness ranging from 13 to 16. Total storage in the waiting queue within the look-ahead horizon capture the spatial requirements for holding interim components for block assemblies. MAR attempts to minimize the area waste by selecting the job most matched to the released area and thus outperforms all other rules. MWMT also performs quite well with no remarkable difference compared to MAR. Other rules are not area utilization oriented and generate relatively higher volume of storage, as expected. Meanwhile, it is found that total storage is very relevant to mean area utilization and decreases with higher utilization throughout the application of all the rules.

Table 2. Performance comparison among different priority rules

		FCFS	SFT	SLACK	MWMT	MAR	EDD	Random	CR
Total Tardiness	*Avg.*	14.1	55.8	12.0	13.5	22.2	33.6	39.2	15.9
(*TT*)	*Dev.*	8.8	18.4	8.5	9.6	17.3	15.2	15.2	11.9
Total Storage	*Avg.*	14014	17324	13955	12974	12840	16061	14414	13394
(*TS*)	*Dev.*	1780	2850	1617	1854	2189	2407	2148	1727
Mean Area	*Avg.*	0.716	0.689	0.718	0.726	0.729	0.699	0.712	0.716
Utilization (*MAU*)	*Dev.*	0.018	0.027	0.013	0.015	0.016	0.026	0.018	0.019

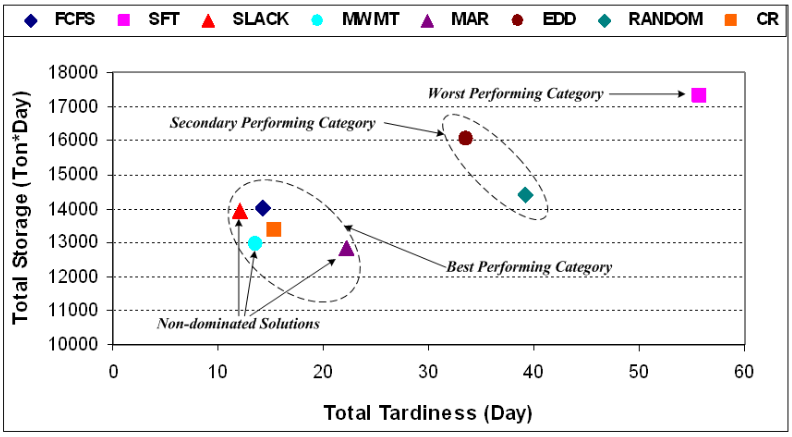

Figure 6. Performance of various priority rules regarding storage and tardiness

Based on the discussion of results presented in Table 2, there is no rule which can achieve best performance regarding all the performance measure. This can be captured in Figure 6, where mean area utilization is ignored since it is directly relevant to total storage. Obviously, rules of FCFS, SLACK, MWMT, CR and MAR generate relatively close performance and are among the best performing category. EDD and RANDOM are seen as the secondary performance category since they are away from the best ones. Although it is reported from the traditional job shop manufacturing that SFT provides satisfactory performance regarding flow time and tardiness, here SFT cannot be found applicable in such operational environment characterized by heterogeneous spatial requirement and synchronized production rhythm. Further observation into the best category, SLACK, MWMT and MAR are not superior to each other but dominate other rules with respect to both performances of storage and tardiness. These three priorities rules construct what is called *Pareto front* in the view of multi-criteria decision making, which provides alternative solution with different preference for managerial objectives. However, the difference in total storage between MWMT and MAR seems very marginal and the difference in total tardiness between MWMT and SLACK does. Hence, MWMT may present a best solution in optimizing the two objectives simultaneously.

Conclusions

In this paper, a discrete-event simulation based mechanism has been presented to assist in effectively scheduling the spatial layout for block assembly in shipbuilding. Rather than generating a deterministic spatial plan, the approach employed aims to provide a look-ahead view for risk hedging by modeling the stochastic properties of production system. In the meanwhile, a heuristic-based algorithm is developed to optimize the location of block assembly by considering the workshop assembly requirement. Future work may include a hierarchical approach for effectively planning and scheduling ship production by combing operational-level simulation models with aggregate-level analytical models.

References

Blackstone, J.H., Phillips, D.T. and Hogg, G.L., 1982. A state-of-the-art survey of dispatching rules for manufacturing job shop operations. *International Journal of Production*, 20(1), 27-45.

Eyres, D.J., 2007. *Ship Construction*. London: Butterworth-Heinemann, Oxford.

Gribsokov, J., 1988. A group technology approach to master scheduling of shipbuilding projects. *The NSAP Ship Production Symposium*, Washington, USA.

Jeong, K.C. and Kim, Y.D., 1998. A real-time scheduling mechanism for a flexible manufacturing system: using simulation and dispatching rules. *International Journal of Production Research*, 36(9), 2609-2626.

Kim, Y.D., Lee, D.H. and Kim, J.U., 1998. A simulation study on lot release control, mask scheduling, and batch scheduling in semiconductor wafer fabrication facilities. *Journal of Manufacturing Systems*, 17(2), 107-117.

Lee, K.J. and Lee, J.K., 1996. A spatial scheduling system and its application to shipbuilding: DAS-CURVE. *Expert Systems with Applications*, 10(3/4), 311-324.

Lee, S., Son, Y.J. and Wysk, R.A., 2007. Simulation-based planning and control: From shop floor to top floor. *Journal of Manufacturing Systems*, 26, 85-98.

Liker, J.K. and Lamb, T., 2000. *Lean manufacturing principles guide* [online]. Available from: http://www.nsrp.org/Project_Information/major_projects/deliverables/ase_91000.pdf [Accessed 10 Jan 2010].

Lodi, A. Martello, S. and Monaci, M., 2002. Two-dimensional packing problem: a survey. *European Journal of Operational Research*, 141, 241-252.

Park, K., Lee, K., Park, S. and Kim, S., 1996. Modeling and solving the spatial block scheduling problem in a shipbuilding company. *Computers and Industrial Engineering*, 30(3), 357-364.

Ryu, C., Shin, J.G. and Lee, J.M., 2008. Development of integrated and interactive spatial planning system of assembly blocks in shipbuilding. *International Journal of Computer Integrated Manufacturing*, 21(8), 911-922.

Sabuncuoglu, I., 1998. A study of scheduling rules of flexible manufacturing systems: a simulation approach. *International Journal of Production Research*, 36(2), 527-546.

Shin, J.G., Kwon, O.H. and Ryu, C., 2008. Heuristic and meta-heuristic spatial planning of assembly blocks with process schedules in an assembly shop using differential evolution. *Production Planning and Control*, 19(6), 605-615.

Son, Y.J., Joshi, S.B., Wysk, R.A. and Smith, J.S., 2002. Simulation-based shop floor control. *Journal of Manufacturing Systems*, 21(5), 380-394.

Spicknall, M.H., 1997. Scheduling methods for ship build strategy development [online]. Available from: http://deepblue.lib.umich.edu/handle/2027.42/1207?mode=full [Accessed 10 Jan 2010].

Wu, S.Y.D. and Wysk, R.A., 1989. An application of discrete-event simulation to on-line control and scheduling in flexible manufacturing. *International Journal of Production Research*, 27(9), 1603-1623.

Zhang, H., Jiang, Z. and Guo, C., 2009, Simulation-based optimization of dispatching rules for semiconductor wafer fabrication system scheduling by the response surface method. *International Journal of Manufacturing Technology*, 41, 110-121.

Appendix

Table. A list of scheduled block assembly for simulation experiment

Project	Block	Ton	Days	Projected Start	Due Date	Size (m×m)	Project	Block	Ton	Days	Projected Start	Due Date	Size (m×m)
1094	C4U3	21	26	6/2/2010	7/13/2010	12×5	1096	HP2DB	87	58	5/3/2010	7/19/2010	10×10
1096	4BF(P)	108	61	5/14/2010	8/2/2010	20×8.5	1096	HC3CI	84	56	5/5/2010	7/18/2010	16×13
1099	PS5P	62	47	5/30/2010	8/2/2010	15×10	1096	HC3SD	110	70	5/28/2010	8/24/2010	17×16
1099	PS5S	62	47	5/28/2010	8/2/2010	18×10	1096	HC1CD	59	43	6/2/2010	8/1/2010	16×9
1095	1BO(P)	62	47	5/28/2010	7/29/2010	12×12	1096	4BF(S)	86	60	5/18/2010	8/7/2010	20×9.5
1099	U3P3	32	34	5/20/2010	7/10/2010	16×7	1099	W4P	22	25	4/30/2010	6/13/2010	9×6
1099	U4P1	15	23	5/3/2010	6/11/2010	9×7	1096	HC4PD	106	75	5/25/2010	8/23/2010	17×15.6
1099	U4P3	11	20	5/17/2010	6/27/2010	9×7	1092	CPD1	41	44	5/6/2010	7/4/2010	12×12
1099	PS3P	58	45	5/27/2010	7/26/2010	17.5×7.5	1099	PS4P	52	51	5/16/2010	7/21/2010	18×8
1094	C4U6	30	32	5/15/2010	7/5/2010	10×5	1095	HP3M	114	67	5/3/2010	7/25/2010	17×17
1095	C1L2	21	27	5/16/2010	7/1/2010	8×5	1096	HC3CD	57	45	4/30/2010	6/29/2010	16×9
1094	C2U7	39	38	5/1/2010	6/24/2010	15×10	1098	CPS	65	41	6/3/2010	8/1/2010	8×5
1094	C2U2	19	28	5/24/2010	7/9/2010	11×4	1098	PLF	22	27	5/7/2010	6/20/2010	6×5
1093	HS4MA	69	50	5/29/2010	8/6/2010	16×12	1098	VH1S	26	28	5/21/2010	9/13/2010	9×5
1093	HP1MA	52	40	5/10/2010	7/7/2010	17×8	1098	VH2P	28	30	5/4/2010	6/18/2010	9×5
1093	HP1I	99	60	5/1/2010	7/15/2010	16×15	1099	CPS	50	41	5/20/2010	7/19/2010	5×5
1092	G0016	22	32	5/11/2010	7/1/2010	5×5	1095	C2L5	24	30	5/17/2010	7/3/2010	6×6
1092	DFW	51	40	6/1/2010	7/30/2010	15×5	1099	PS3A(P)	25	30	5/19/2010	7/6/2010	10×6
1092	DFDP	31	32	5/24/2010	7/14/2010	16×6	1095	C1L5	24	30	5/19/2010	7/9/2010	6×6
1094	C4U7	32	33	5/21/2010	7/12/2010	12×5	1095	HS3I	53	48	5/6/2010	7/12/2010	8×7