DAS: Intelligent Scheduling Systems for Shipbuilding

Jae Kyu Lee*, Kyoung Jun Lee*, June Seok Hong*, Wooju Kim*, Eun Young Kim, Soo Yeoul Choi**, Ho Dong Kim**, Ok Ryul Yang**, and Hyung Rim Choi*****

* Dept. of MIS, Korea Advanced Institute of Science and Technology, 207-43, Cheongryang, Seoul 130-012, Korea.
jklee@msd.kaist.ac.kr, leekj@msd.kaist.ac.kr, jshong@msd.kaist.ac.kr, wjkim@msd.kaist.ac.kr
** System Planning Team, Daewoo Heavy Industries Ltd, Okpo. 1, Ajoo-Dong. Jangseungpo, Kyungnam, Korea.
*** Dept. of MIS, Dong-A University, 840, Hadan-dong, Saha-gu, Busan 604-714, Korea.

Abstract

Daewoo Shipbuilding Company, one of the largest shipbuilders in the world, has been much troubled with planning and scheduling its production process for a long time since its establishment. To solve the problems, Korea Advanced Institute of Science and Technology (KAIST) and Daewoo have been jointly performing the DAS (DAewoo Shipbuilding Scheduling) Project for three years from 1991 to 1993. For integrating the scheduling expert systems for shipbuilding, we employed the hierarchical scheduling architecture. To automate dynamic spatial layout of objects in various areas of the shipyard, we developed spatial scheduling expert systems. For reliable estimation of man-hour requirements, we implemented the neural network based man-hour estimator. In addition, we developed the Panelled Block Assembly Shop Scheduler and the Long Range Production Planner. For this large-scale project, we devised the phased development strategy consisting of three phases such as vision revelation, data dependent realization, and prospective enhancement. The DAS systems successfully launched in January 1994 and are being actively used as indispensable systems in the shipyard resulting in significant improvement in productivity and reporting visible and positive impacts in many ways.

Problem Description: The Company's History Struggling with Scheduling

Daewoo Shipbuilding Company is one of the largest shipbuilders in the world. Its main products are VLCC (Very Large Crude Oil Carrier) and container ship. It also involves heavy machinery, plant and automobiles that make minor contribution to the company sales. Its shipbuilding yard has three docks. One is the largest dock in the world capable of manufacturing one million ton VLCC and has a Goliath crane which can hold up to 900 tons. Employees are twelve thousand and the sales in 1993 were 2 billion dollars. Shipbuilding is the make-to-order manufacturing and takes long time to complete the products. It includes so many complicated processes from an order to the final delivery and requires large amount of working capital. It is very difficult to manage the resources including material, human, facilities and information. Therefore, the scheduling and control of shipbuilding plant is a very complex task and a nightmare. The company has struggled for an effective scheduling to achieve total optimization since its establishment. Poor scheduling keeps workers waiting for the prerequisite subassemblies, causes fluctuation of work loads resulting in expensive overtime works, and causes delay in delivery. The company has attempted various project management software such as PROJACS, VISION and X-PERT as well as in-house development with conventional programs, but all these efforts failed because they couldn't reflect scheduler's knowledge flexibly considering the complicated interrelated factors. In addition, no software could support the dynamic spatial layout even though the spatial resources with material handling equipment like cranes are bottlenecked resources. To develop the integrated scheduling expert systems overcoming these problems, KAIST and Daewoo have been jointly performing the DAS (DAewoo Shipbuilding Scheduling Expert Systems) Project for three years from 1991 to 1993. The DAS systems successfully launched in January 1994 and have been used as indispensable systems in the shipyard resulting in the significant improvement in productivity and reporting the visible and positive impact in many ways.

Application Description

The DAS project adopted various expert systems technologies to cope with the scheduling task. Key sub-systems in DAS project are as follows.

DAS-ERECT: Erection Scheduler at Docks

DAS-CURVE: Curved Block Assembly Shop Scheduler

DAS-PANEL: Panelled Block Assembly Shop Scheduler

DAS-MH: Neural Network based Man-hour Estimator

DAS-LPP: Long-term Production Planner

DAS-ERECT: Erection Scheduler at Docks

The initial shipbuilding process begins with block division. Usually, a large tanker is divided into hundreds of sized blocks as shown in Figure 1. These can be categorized into two types: flat-bottomed, which are assembled in the Panelled Block Assembly Shop(PBS), and curved-bottomed, which are assembled in the Curved Block Assembly Shop(CBS). Multiple blocks may be assembled into larger ones (called super-blocks) in the Pre-Erection Shop(PES) to reduce the assembly time needed at the main erection dock. At the main docks a number of blocks and super-blocks are constructed ('erected') using a very large Goliath crane. In a shipbuilding plant, there are usually one or two dry docks for erection and corresponding pre-erection shops as shown in Figure 2.

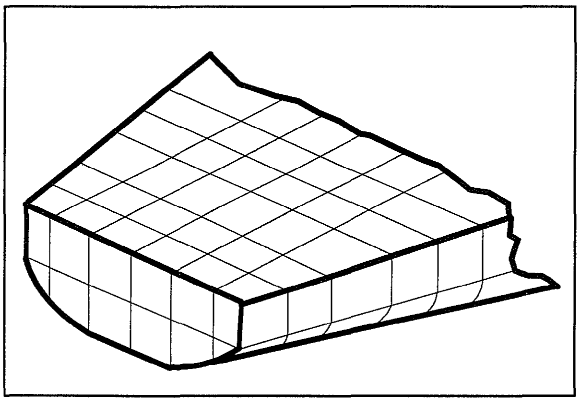

Figure 1. A Ship and its Partition into Blocks

The important characteristics of erection scheduling include:

Sequential Erection at Each Dock: Blocks and super-blocks in a dock have to be erected one at a time because there is only one Goliath crane at each dock. However, there may be multiple docks in some plants.

Large Search Space: Since the schedulers have to consider multiple ships, each composed of 400 ~ 500 blocks, a manual search for the best schedule is beyond mental processing capacity. Thus, there is a need to develop a proper search method.

Technical Knowledge for Erection Sequencing: The technical knowledge which restricts the erection sequences is available.

Utilization of Resources: Key resources in shipbuilding comprise human workforces, cranes and space. To utilize these resources effectively, a schedule should be established to balance the loads among PBS, CBS, Pre-erection shops and docks.

Hierarchical System Architecture for the Shipbuilding Scheduling

For erection scheduling at the dock, we employed the hierarchical scheduling architecture (Lee, Suh, & Fox 1993) and developed the constraint directed graph search technique (Lee et al. 1994). In the hierarchical architecture, detailed schedules for the lower level assembly plants are delegated to the individual plant's schedulers (DAS-CURVE and DAS-PANEL in this case) as far as the requirements from the higher level scheduler (DAS-ERECT) are satisfied(Figure 3). However, if the lower level scheduling is impossible, the higher level scheduler attempts to adjust the original requirements. In the constraint directed graph search, we amalgamated the notions of graph expansion and constraint directed pruning into an algorithm.

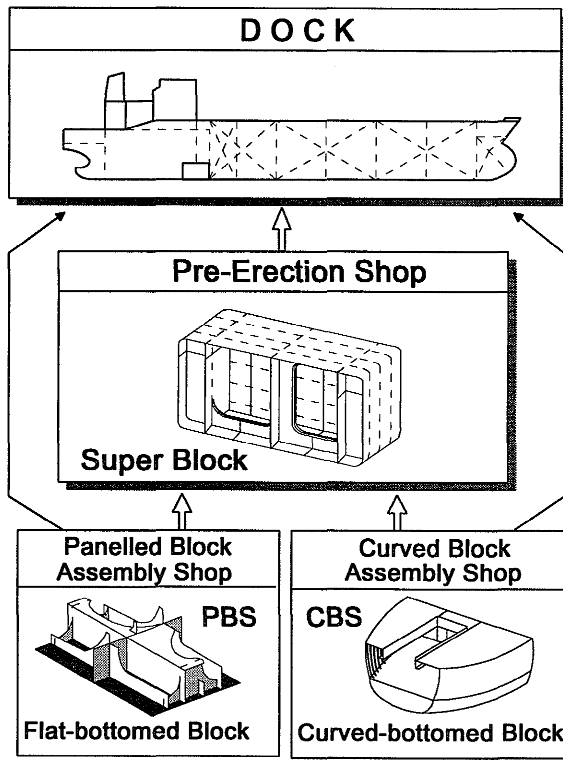

Figure 2. Typical Shipbuilding Procedure

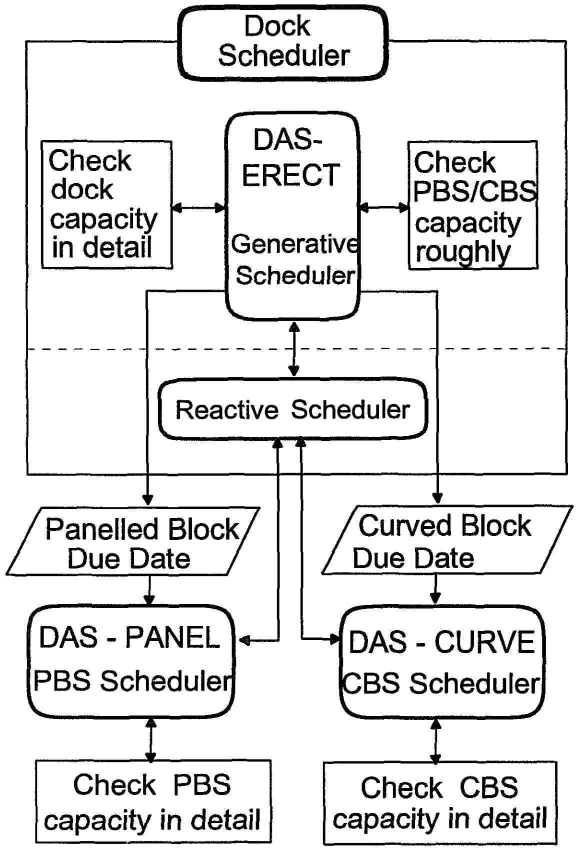

Figure 3. Hierarchical Architecture of Shipbuilding Scheduling

To reduce complexity during the generative scheduling phase and to improve the robustness during reactive scheduling phase, a hierarchical architecture as shown in Figure 3 has been adopted. In this architecture, part of overall scheduling is delegated to the lower level schedulers - PBS and CBS schedulers - which schedule and control their respective work areas. In our model, each lower level scheduler can handle its own work area independently as long as it satisfies the requirements imposed by the higher level dock scheduler.

Block Assembly Due Date Generation by Dock's Generative Scheduler: The Dock's Generative Scheduler determines the due date of each block's assembly. In this stage, the total capacities of lower-level assembly shops like PBS and CBS are considered. Then the scheduler requests the relevant lower level schedulers to deliver the blocks by these due dates.

Distributed Scheduling and Controlling by PBS and CBS Schedulers: Each scheduler at PBS and CBS schedules its own work area according to the requested due dates as well as the spatial and manpower constraints. When a certain block cannot be assembled in time, the scheduler reports the trouble to the higher level dock scheduler to obtain assistance in resolving the inconsistency. Then the due date of a block may be adjusted, or overtime may be required.

Constraint Directed Graph Search

In order to find the best erection sequence at the dock, a new search procedure has been developed. The first observation is that graph search will accomplish this purpose. The graph search expands nodes, and possibly selects a proper node possibly using an evaluation function(Nilsson, 1980). However, the application of a pure graph search is not appropriate in this case because measurement of multiple realistic evaluation functions is almost impossible. The second observation is that there are precedence constraints between nodes due to the technical reasons. To accommodate the graph expansion framework with the constraint considerations, we propose an algorithm called Constraint Directed Graph Search (CDGS).

Algorithm of CDGS

To develop an algorithm for the CDGS procedure, let's consider the case of a single vessel. This algorithm can be extended to multiple vessel and multiple dock cases.

Notations

N_i: a node *i* (which implies a block in this case), $i = 1, \ldots, m$

$WHOLE = \{ N_i \mid i = 1, \ldots, m \}$

KL: potential list of keel laying blocks which are the first to be laid for each vessel

PATH: list of scheduled nodes

POTENTIAL: union of adjacent nodes to the ones in current *PATH* excluding those in the *PATH* itself

ERECTABLE: list of nodes that satisfy the technical constraints among the nodes in the current *POTENTIAL*

i-th(LIST): *i*-th node in the *LIST* list

Algorithm

Phase 1: Initialization

Step 0. Acquire the required data and constraints.

Step 1. Put all the nodes to be erected in the *WHOLE list*, and initialize the empty lists.

$WHOLE \leftarrow \{N_1, \dots, N_m\}$

$PATH \leftarrow \phi$

$POTENTIAL \leftarrow \phi$

$ERECTABLE \leftarrow \phi$

Phase 2: Selection of Keel Laying Block

Step 2. Generate the candidate keel laying blocks and store them into *KL*. In this step the potential keel-laying block selection constraints are used.

$KL \leftarrow$ Keel_Laying_Constraints (*WHOLE*)

Step 3. Select a keel-laying block based on the selection strategy. Put the selected keel laying block in the *PATH list.*

$N_k \leftarrow$ Select_a_Keel_Laying_Block (*KL*)

$KL \leftarrow KL - \{N_k\}$

$PATH \leftarrow PATH + \{N_k\}$

$WHOLE \leftarrow WHOLE - PATH$

Phase 3: Graph Expansion

Step 4. Expand *POTENTIAL* of current *PATH* by adding adjacent nodes of N_k while not in the *PATH.*

$POTENTIAL \leftarrow POTENTIAL \cup$ {Adjacent_Node(N_k) - *PATH*}

Step 5. If the $POTENTIAL = \phi$ and $WHOLE = \phi$, all nodes are erected appropriately. Stop.

If $POTENTIAL = \phi$ while $WHOLE \neq \phi$, the search procedure has problems. Stop with an error message.

Step 6. Among the nodes in *POTENTIAL*, select the nodes which also satisfy the technical constraints.

$ERECTABLE \leftarrow$ Technical_Constraints (*POTENTIAL*)

Step 7. If $ERECTABLE = \phi$, backtrack.

$N_z \leftarrow$ *last_th(PATH)*

$PATH \leftarrow PATH - \{N_z\}$

$WHOLE \leftarrow$ *Cons(WHOLE, N_z)*

$POTENTIAL \leftarrow POTENTIAL - \{N_z\}$ - {Added_Potential_by (N_z)}

$ERECTABLE \leftarrow$ Technical_Constraints (*POTENTIAL*)

$N_k \leftarrow$ Evaluate_and_Find_the_Best (*ERECTABLE*)

Then, go to Step 4. Otherwise, continue the next step.

Step 8. Select the best node to be erected from *ERECTABLE*. Update each list.

$N_k \leftarrow$ Evaluate_and_Find_the_Best (*ERECTABLE*)

$PATH \leftarrow$ Cons(*PATH*, N_k)

$POTENTIAL \leftarrow POTENTIAL - \{N_k\}$

$WHOLE \leftarrow WHOLE - \{N_k\}$

Go to Step 4.

Knowledge Representation for CDGS

The knowledge base used in the CDGS is composed of object oriented data and constraints. Objects represent the hierarchical structure of blocks and super-blocks, and the resources utilized in each shop (PBS, CBS, PE shop, dock shop, etc.) as follows:

{{ **5060-203**

IS-A: BLOCK
SHOP: PBS
PART-OF: MID-BODY
STRUCTURAL-PART-OF: BOTTOM

"Relative Position"
REAR: 5060-232 5060-222
LEFT: 5060-6A0
RIGHT: 5060-6B0
FRONT: 5060-204
UP: 5060-442 5060-6C0 5060-4B0
DOWN:

"Estimated Durations (days) and Manhours"
BLOCK-ASSEMBLY-DURATION: 9
BLOCK-ASSEMBLY-MANHOUR: 346
DOCK-SETTING-DURATION: 1
DOCK-SETTING-MANHOUR: 57
DOCK-FITTING-DURATION: 4
DOCK-FITTING-MANHOUR: 73
DOCK-WELDING-DURATION: 3

DOCK-WELDING-MANHOUR: 73

"Generated Schedules"

ASSEMBLY-SST: 92/01/03
ASSEMBLY-SFT: 92/01/11
DOCK-SETTING-SST: 92/02/09
DOCK-SETTING-SFT: 92/02/09
DOCK-FITTING-SST: 92/02/10
DOCK-FITTING-SFT: 92/02/13
DOCK-WELDING-SST: 92/02/11
DOCK-WELDING-SFT: 92/02/13 }}

* SST : Scheduled Start Time
SFT : Scheduled Finish Time

Ten types of technical constraints are acquired and can be grouped into three classes. Descriptive examples from each class are as follows:

General Technical Constraints: 'A keel laying block should be selected among the non-side and bottom blocks of engine-room or mid-body part.'

Constraints on Partial Sequence: 'All the blocks should satisfy the structural stability condition. Therefore the transverse sequence of the mid-body must satisfy the following sequence: Bottom → Bulk-Head → Side-Shell → Deck.'

Constraints on Precedence Relationship: 'The block that needs supporting facilities must be erected after having erected the base blocks on which the facilities should stand.'

For explanatory purpose, use three erected blocks, as shown in Figure 4. Suppose *PATH* = { PA, PB, PC }. Then there are ten potentially erectable blocks. Therefore, *POTENTIAL* = {P1, P2, ... , P10}. The potentially erectable blocks and their positions are shown in Figure 5. We can derive *ERECTABLE* list from the *POTENTIAL* list by considering technical constraints such as *structural stability*. By applying this constraint, suppose the *ERECTABLE* list has become {P1, P2}. To select one block, we evaluate blocks P1 and P2. Use two criteria: the resource utilization level of lower-level assembly shops and the earliest erection start time at dock. Suppose that blocks P1 and P2 in the current *ERECTABLE* list are flat-bottomed, and they are evaluated by the utilization level of the PBS. If the block is to be erected on the 7th day at the dock, the block P2 should be preferred as the next block to be erected. If P2 is selected, the *PATH* list becomes {PA, PB, PC, P2}. The above procedure will be repeated until all blocks are erected. The Figure 6 shows an erection network scheduled by DAS-ERECT.

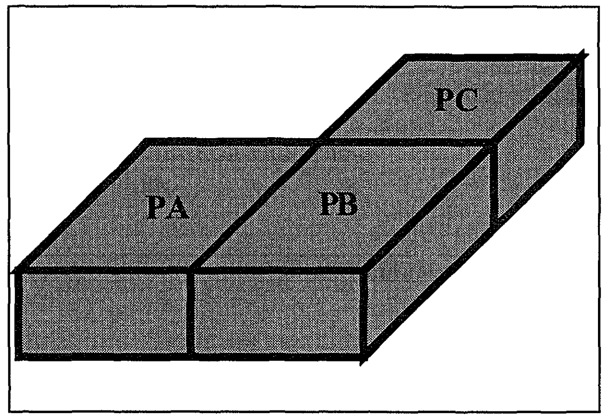

Figure 4. Partially Erected Blocks

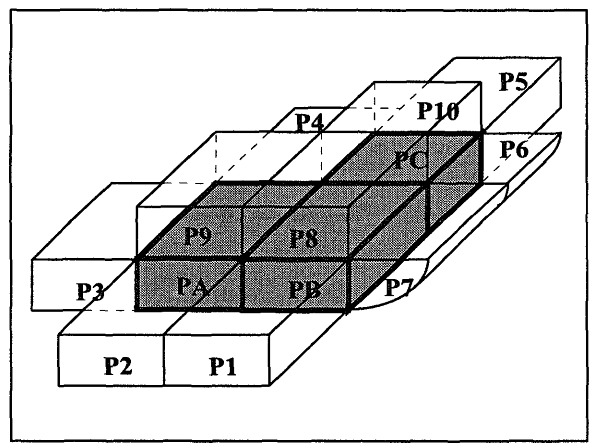

Figure 5. Spatial Position of Potentially Erectable Blocks

Spatial Scheduling in Shipbuilding

In the shipbuilding plants which handle the heavy and bulky blocks, it is necessary to employ expensive material handling equipment like cranes and work plates. Since the space equipped with such facilities is usually limited and bottlenecked, the scheduling needs to consider the spatial resources as well as traditional ones like manpower and machines. We call this kind of scheduling a spatial scheduling in this project. As the term implies, the spatial scheduling deals with the optimal dynamic spatial layout schedules. In a shipyard, spatial scheduling problems occur frequently in various working areas like erection docks, pre-erection shops (that assemble the blocks into larger super blocks), and

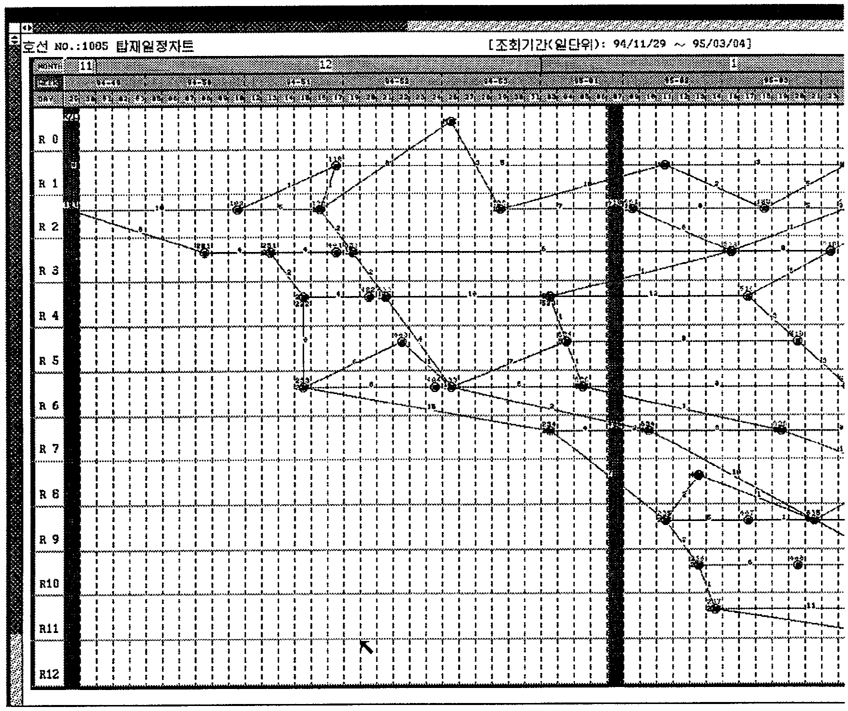

Figure 6. An Erection Network Scheduled by DAS-ERECT

block-assembly shop, etc. So far, human schedulers have carried out the spatial scheduling manually without any automated aids. Even though human experts have long experience in spatial scheduling, it takes long time and heavy efforts to produce a satisfactory schedule because of its huge search space required to consider blocks' geometric shapes. For example, spatial scheduling for six months is beyond the scope of human's mental capacity, so it has been impossible to build such a large scale spatial schedule in advance. Nevertheless, the spatial resource tends to be a critical resource in shipbuilding. Therefore, automation of the spatial scheduling process has been a critical issue for the improvement of productivity in the shipbuilding plants and the total integration of scheduling expert systems.

In Daewoo, there have been some tries to solve the spatial scheduling by themselves. One approach was a simple spatial scheduling system assuming the shape of blocks can be approximated to rectangle. But the field schedulers rejected it because the approximation sacrifices spatial utilization too much. DAS-CURVE assumes the approximation of the blocks' shape to polygons which users agreed. Another approach failed is the interactive spatial scheduling support system that helps human scheduler by providing the graphic user interface to them. The interactive system was not effective because it couldn't automate the tedious spatial scheduling process and reduce the information burden and scheduling time. It could only replace the paper and pencil with the computerized interface.

The objectives of spatial scheduling may vary somewhat, depending on the nature of a given plant. In general, however, spatial scheduling systems pursue the objectives of due-date satisfaction, maximal utilization of spatial and non-spatial resources, and minimization of waiting times for work-in-process and final product inventories.

Typical constraints include crane capacity, man-hour availability, assembly due date, precedence between associated assemblies, physical adjacency of coupled objects for operational efficiency, minimum required distance between blocks, and maximum acceptable waiting times for completed and work-in-process blocks.

Typical necessary input data include jobs with due-dates and their constituent activities, two-dimensional geometric spatial objects of the activities, required processing times for each activity, and spatial shapes of work plates.

In the shipbuilding domain, the shapes of most objects tend to be convex polygons like triangles, rectangles, or trapezoids. Some blocks may have some local concavity.

However, in most cases, the local concave space is not usable by other objects. Therefore, they can be approximated as convex polygons.

In shipbuilding, the orientation of an object is prefixed to four alternative orientations (0°, 90°, 180°, and 270°) to ensure stable crane operations.

Search Space in Spatial Scheduling

To find the feasible locatable positions of an object a_i within a workplate W that do not overlap a scheduled object b_j, we can adopt the notion of configuration space (Lozano-Perez 1983; Zhu & Latombe 1991). Configuration space is the space through which the reference point of an object (a robot, for example) with fixed orientation can possibly pass without colliding with present obstacles. In our research, there are two kinds of configuration spaces: *Obstacle Avoiding Space* and *Inner Locatable Space.* The *Obstacle Avoiding Space* $S(a_i|B)$ is a space where the reference point of an object a_i can be located without colliding with the already located objects which are regarded as obstacles. *Inner Locatable Space* $S(a_i|W)$ is the space where the reference point of an object a_i can be located within the boundary of a work area W. Thus the *Feasible Locatable Space* $S(a_i|B, W)$ can be derived by intersecting the above two spaces: $S(a_i|B, W) = S(a_i|B) \cap S(a_i|W)$.

Figure 7 illustrates the spaces where the object a_i is to be located within W, on which two objects b_1 and b_2 are already located.

Since the Feasible Locatable Space is a continuous space, it is impossible to find all the points in it. To extract a set of meaningful discrete points out of the continuous space, we define *Distinctive Locatable Point Set* $D(a_i|B, W)$, which consists of the vertices of the Feasible Locatable Space.

Theoretically speaking, the Distinctive Locatable Point Set does not guarantee finding an optimal location (Lee & Lee 1995). However, the points have empirically provided very satisfactory locations with the advantage of computational efficiency.

The Distinctive Locatable Points can be classified into four categories:

$D_O(a_i|B, W)$: Union of the feasible vertices of each Obstacle Avoiding Space

$D_I(a_i|B, W)$: Feasible vertices of the Inner Locatable Space

$D_{OI}(a_i|B, W)$: Union of the feasible intersection points between the boundaries of each Obstacle Avoiding Space and the Inner Locatable Space

$D_{OO}(a_i|B, W)$: Union of the feasible intersection points between the boundaries of each Obstacle Avoiding Space

These points are illustrated in Figure 7: three $D_O(a_i|B, W)$ marked 'triangle', three $D_I(a_i|B, W)$ marked 'square', six $D_{OI}(a_i|B, W)$ marked 'circle', and two $D_{OO}(a_i|B, W)$ marked 'diamond'.

The categorized points can be used to reduce the search space by identifying a special category of the Distinctive Locatable Point Set contingent to the situation.

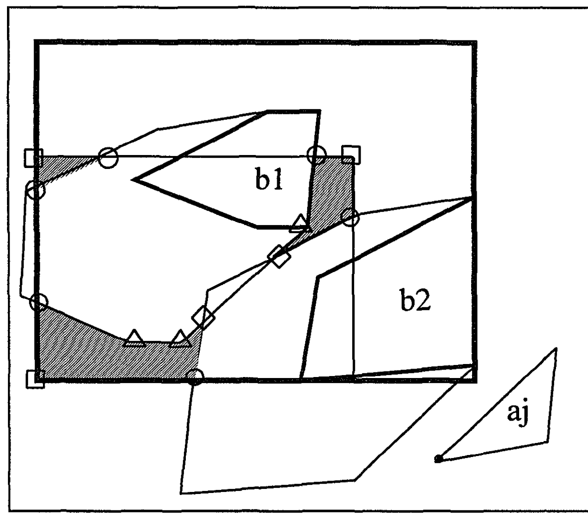

Figure 7. Feasible Locatable Space and Distinctive Locatable Points

Search in the Distinctive Locatable Point Set

Since most of layout problems are NP-complete, we propose four positioning strategies which can be effectively applied contingent to the situation:

1) Maximal Remnant Space Utilization Strategy
2) Maximal Free Rectangular Space Strategy
3) Initial Positioning Strategy
4) Edging Strategy

Since each strategy has its own merits depending on the situation, we synthesize a composite positioning algorithm which can apply an adequate positioning strategy contingently. Key issues in composite positioning are the identification of situation, reduction of search space, and selection of effective strategy. In this study, we identify four types of situations depending on the existence of already-located objects, the attempted location (near corner, edge, or other object), and the shapes of the objects. The strategies are by no means complete. However, we have empirically verified that these four situation types can still quite effectively capture the possible situations.

Composite Positioning Algorithm

Compute $D(a_i|B,W) = [d_1, d_2, \ldots, d_k, \ldots, d_z]$.

$CS := D(a_i|B,W)$ /* CS=a set of Candidate Solutions */

IF $CS = NULL$

THEN /* a_i cannot be allocated in W without changing the current schedule */

Start Backtracking.

ELSE

IF *there is no located objects yet*

THEN *Reset* $CS := D_I(a_i|W)$

IF a_i *is not rectangular*

THEN *Apply the Initial Positioning Strategy*

ELSE /* a_i is rectangular */

THEN *Select a point arbitrarily*

ELSE /* there is no located objects yet*/

IF *most* b_j*'s are rectangular OR near rectangular*

THEN *Apply the Maximal Free Rectangular Space Strategy*

ELSE /* most b_j's are not rectangular */

IF $CS := D_{OI}(a_i|B,W)$

THEN *Apply the Edging Strategy*

ELSE

THEN *Apply the Maximal Remnant Space Utilization Strategy*

Backtracking and Adjustment in Spatial Scheduling

If we cannot find a feasible schedule for a certain day, we have to backtrack to adjust the current spatial layout, the starting times of already-scheduled activities, and/or the resource commitment level. Some of the backtracked adjustment could have been avoided if we could have looked ahead to what is needed to be located over the next several days. However, obtaining information about the precise impact of these future objects is almost as expensive as the scheduling itself. Therefore, we adopt the backtracking and adjustment strategy.

For the shipbuilding domain, we utilized the following six types of adjustment:

1) Work Plate Re-Selection

2) Intra-Plate Spatial Adjustment

3) Inter-Plate Spatial Adjustment

4) Intra-Plate Temporal Adjustment

5) Intra-Plate Spatiotemporal Adjustment

6) Inter-Plate Spatiotemporal Adjustment

Principles for the spatial scheduler to follow in sequencing the application of the above adjustment types can be summarized as follows:

1) Monitor the status of the free rectangular space of each work plate as a measure of spaciousness. For a new object, assign a work plate according to this criterion as long as the special constraints for the new object are satisfied. However, if the shape of the free space turns out to be unsuitable for accommodating the particular shape of a new object, then re-select the work plate if other sufficiently spacious options are left.

2) Apply the spatial adjustment (including the work plate re-selection) before applying the temporal adjustment because the temporal adjustment invokes a trade-off between the tardiness of new activity and the work-in-process inventory of allocated activities. However, spatial adjustment does not incur such a cost as long as the constraint of operational adjacency among certain blocks is not violated.

3) Apply the intra-plate adjustment before applying the inter-plate adjustment because the intra-plate search is intuitively a more efficient option.

4) Apply the purely spatial and/or temporal adjustment strategy first before applying the spatiotemporal adjustment. This prioritization is intuitively valid because the composite spatiotemporal adjustment necessitates greater computation than the single adjustment strategy.

The sequencing rules in applying the adjustment strategies do not have to be sequential. However, the computational effort of identifying the optimal adjustment strategy is very expensive, as is the look-ahead strategy. Therefore, it is desirable to have a reasonable predetermined default sequence. In this study, the priority of adjustment strategies is incorporated in the adjustment algorithm based on the above rationale.

Another default scheduling strategy adopted here with respect to the starting time of activities is the ***latest starting time strategy***: "Assign the activity to be finished one day before the due date to avoid the wait of work-in-process inventory. However, if this is not possible to achieve, try to minimize the waiting time."

The contingent choice of the proper adjustment strategy is related to the cause of scheduling failure and the scheduling status. For example,

IF the Cause of the Scheduling Failure is deficiency of manpower
THEN Skip the Intra-Plate Spatial Adjustment because it will not end successfully.

IF the Space Utilization of a work plate is relatively low,
THEN The Intra-Plate Spatial Adjustment is more effective than the Intra-Plate Temporal Adjustment

IF the Space Utilization of a work plate is relatively high
THEN The Intra-Plate Temporal Adjustment is more effective than the Intra-Plate Spatial Adjustment

IF the Space Utilization of every work plate is very high
THEN Skip the Inter-plate Spatial Adjustment

DAS-CURVE: Curved Block Assembly Shop Scheduler

DAS-CURVE is a representative spatial scheduling expert system(Lee & Lee 1992; Lee, Lee & Choi 1995) for the curved-bottomed block assembly shop, applying firstly UNIK-SPACE. The system generates the spatial schedule of assembling blocks meeting the due-dates imposed by DAS-ERECT. The block assembly shop has about 15 work plates with cranes to lift blocks and subassemblies. The resources are limited by the availability of the spatial work plates, as well as the non-spatial resources of manpower and cranes. Figure 8 illustrates an output screen of DAS-CURVE showing a snapshot spatial layout status of the eight work plates in the shop for a day, while Figure 9 illustrates the dynamic spatial layouts of a work plate during an indicated time interval.

As the term 'spatial' scheduling implies, the visual interactive scheduling is an essential feature for the

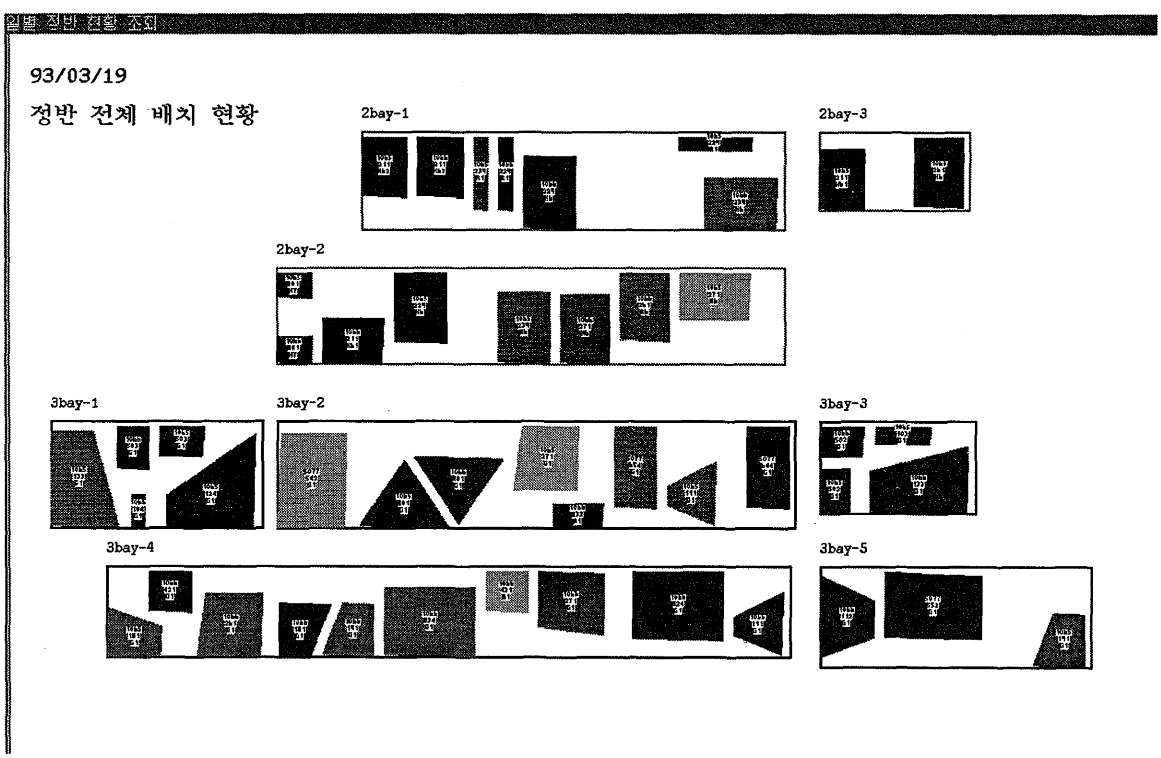

Figure 8. An Output Screen Showing a Snapshot of Spatial Layout in DAS-CURVE

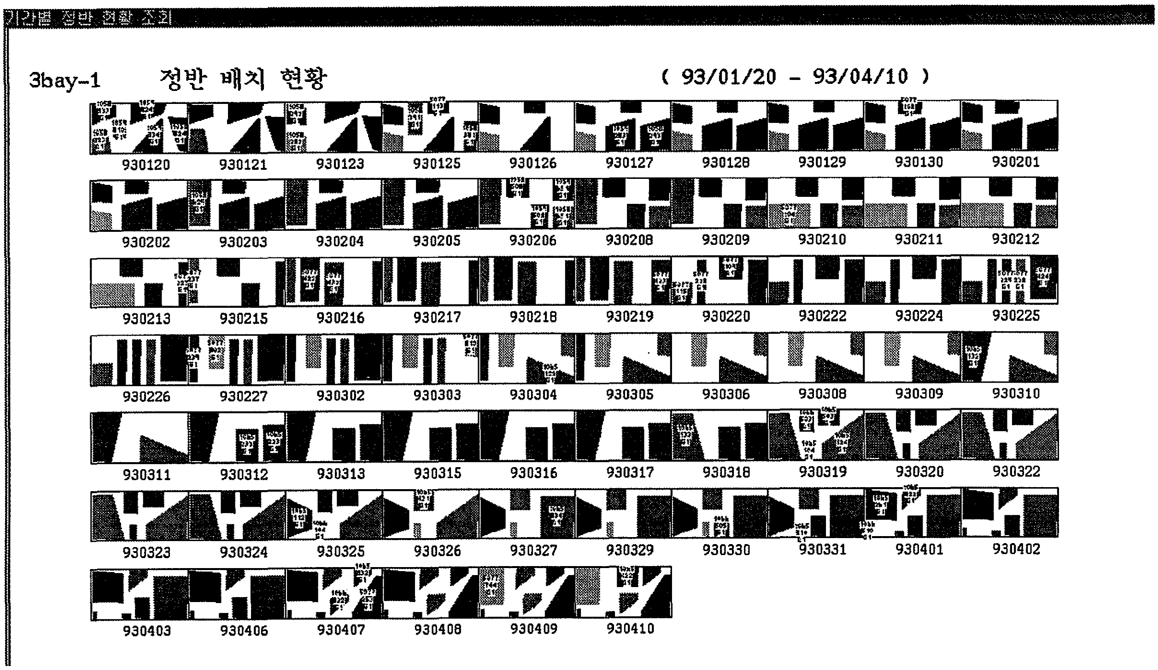

Figure 9. An Output Screen Showing a Dynamic Spatial Layout of a Work Plate in DAS-CURVE

scheduler's initiative. Therefore, DAS-CURVE is equipped with the mouse-based graphic user interface. To maintain consistency between the user's visual interactive input and the invisible constraints, the reactive scheduling capability should work behind the screen, usually adopting the adjustment methods described earlier.

Spatial Scheduling in Other Systems

In erection scheduling, the spatial scheduling problem occurs in the pre-erection shop near the docks. The pre-erection shop has many similarities with the curved-bottomed block assembly shop, but has some spatial layout constraints which do not exist in the curved-bottomed block assembly shop. These differences result from the differences in the technical constraints between the two shops. The spatial scheduling module reflecting these features is imbedded in the DAS-ERECT.

The long range planning of a shipyard should consider the spatial capacities of docks because the sizes and the shapes of ships are diverse. Especially when there are hundreds of product mix candidates, automated checking of spatial availability is essential because there exist various spatial constraints on the layout of ships due to the technical constraints during the erection process. DAS-LPP has the dynamic spatial layout module using UNIK-SPACE(Lee, Lee & Choi 1994).

To implement the spatial scheduling methodology, many computational geometric algorithms are used, such as the 'Point-in-Polygon' algorithm (Preparata & Shamos 1985) to determine whether a point is in a convex polygon, the Polygon-Intersection algorithm (O'Rourke etc. 1983) to compute the intersection of two convex polygons, and the Polygon Setsum algorithm (Lozano-Perez 1983) to compute the setsum of two convex polygons. In addition to these known algorithms, various specialized computational geometric algorithms are devised for spatial scheduling, such as those for enlarging polygons to the extent of the safety distance and computing feasible intersection points among polygons.

DAS-PANEL: Panelled Block Assembly Shop Scheduler

DAS-PANEL(Hong, Kim & Lee 1993) is a scheduling expert system for the panelled block assembly shop in which blocks and their subassemblies are welded on the assembly lines and adjacent off-lines, respectively. Since the block size and compositions of subassemblies are diverse, the scheduling should encompass the detailed

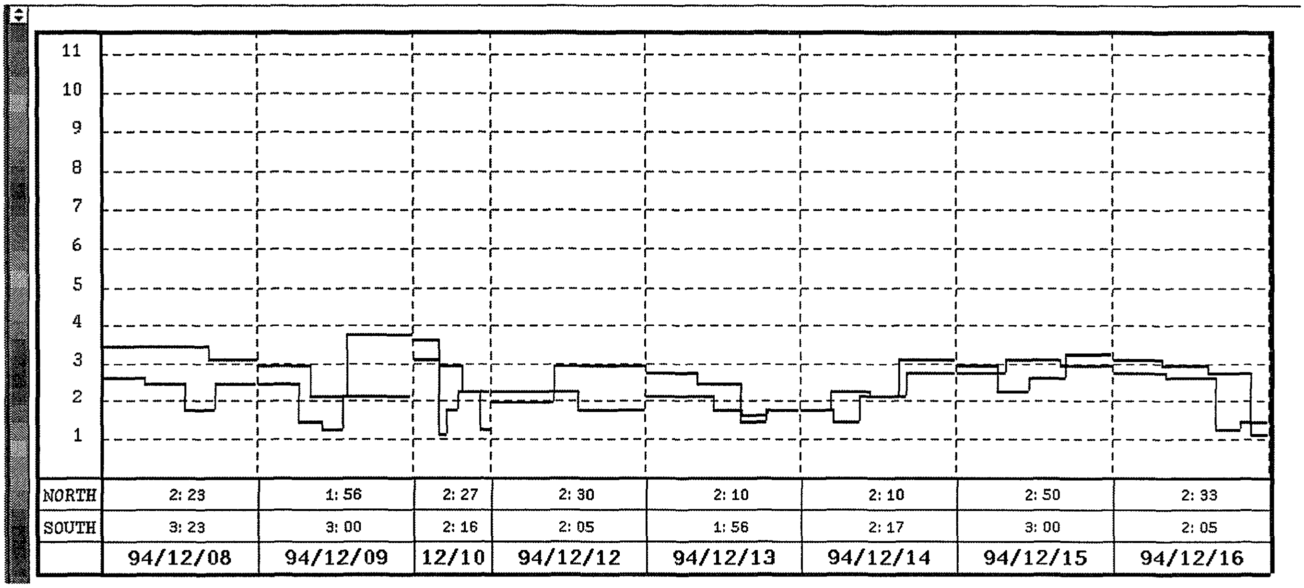

Figure 10. An Output Screen Showing Dynamically Changing Cycle Time in DAS-PANEL

scheduling of subassemblies as well as the main assembly line scheduling of blocks. DAS-PANEL has two characteristics: (1) Since the blocks made in the shop are one-of-a-kind, the speed of the assembly line should not be fixed in advance. To improve the productivity of the assembly line, it is necessary to generate a schedule which can dynamically change the cycle time according to the characteristics of blocks. (2) Its objective is to minimize not only the assembling time of blocks on the main line, but also the waiting times of subassemblies and blocks. DAS-PANEL was built using a typical forward chaining rule-based tool UNIK-FWD written in C++. Figure 10 shows dynamically changing cycle time generated by DAS-PANEL

DAS-MH: Neural Network based Man-hours Estimator

To establish reliable scheduling expert systems, the estimation of accurate man-hour requirement for each assembly is a prerequisite. To supply the required welding man-hours inputs of each one-of-a-kind block for DAS scheduling systems, we adopted the artificial neural network as an estimator (Lee & Kim 1994).

To build a reliable and efficient neural network model, we take the following research procedure.

1) Select candidate input variables: Four categories of variables are selected with possible values for each variable.
2) Eliminate unnecessary variables: Highly correlated redundant cardinal variables are filtered out by stepwise regression as a preprocessor.
3) Train and test the network with and without the preprocessing.
4) Compare the estimation performance by the neural network with one by the regression analysis.

First of all, we have selected 4 categories of variables that influence the welding time. They are candidate input variables of the neural network model.

1) Ship Type

Usage: Select one of the following values.

VLCC(Very Large Crude oil Carrier)

OBO(Ore Bulk Carrier)

COT(Crude Oil Tanker)

B/C(Bulk Carrier)

CONT(Container)

P/C(Product Carrier)

etc.

Dead Weight (which means the weight that the ship can carry)

2) Block Type

Locus: Select one of the following values.

Side-Shell

Bulbus-Low

Engine-Room-Side-Upper

Fore-Deck

Longitudinal-Bulk-Head

Transverse-Bulk-Head

Transverse-Non-Corrugated-Bulk-Head

Mid-Ship-Bottom

etc.

Panelled or Curved Bottom

Single or Double Shell

3) Block's Physical Characteristics

Welding Length

Joint Length

Block's Weight

4) Shop Type

Assembly Shop: Select one of the following values.

3DS(Three Dimensional Shop)

BOS(Building Outfitting Shop)

PBS(Panelled Block Shop)

A-7(Area Code No.7)

Indoor or Outdoor for each shop

In total, we have 10 italicized candidate input variables: 4 cardinal and 6 nominal scale variables.

To eliminate unnecessary variables, we have adopted the stepwise regression approach with four cardinal variables: dead weight, welding length, joint length, and block's weight.

We expected that multicollinear variables better be dropped off to keep the model robust. In this manner, the joint length and block's weight are finally selected. Thus 8 variables are eventually selected as the inputs to the neural network. Figure 11 shows a configuration of the neural network developed using a neural network developer, UNIK-NEURO.

To confirm whether the neural network model significantly outperforms other possible approaches, we have compared with two approaches: multiple regression

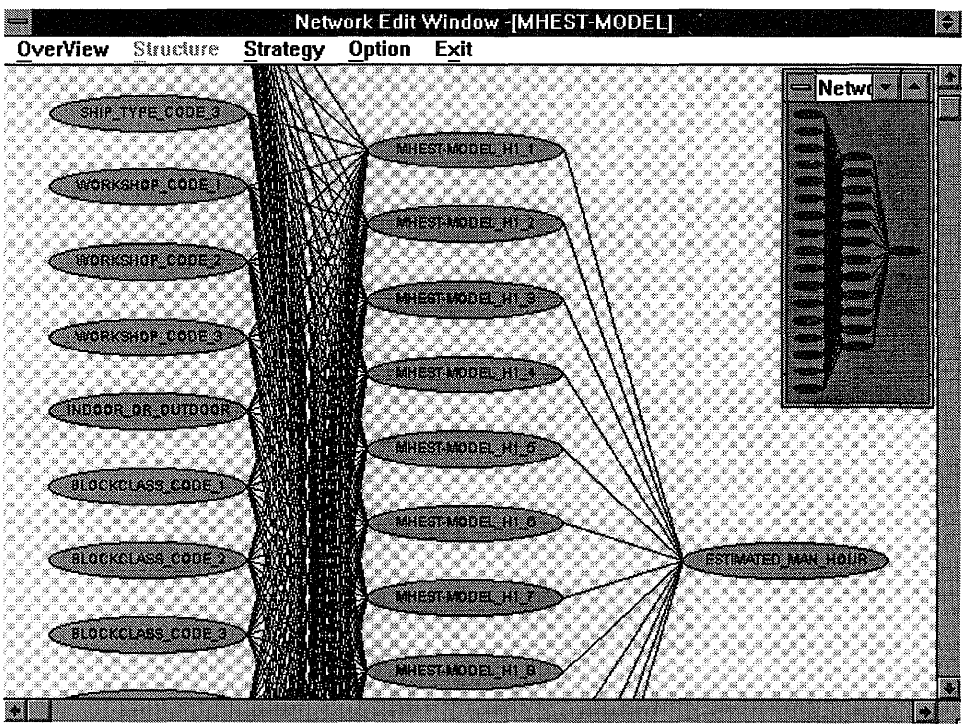

Figure 11. A Neural Network Configuration for Manhour Estimation

model and simple regression through origin. The reason why we compare with the simple regression through origin is that this method has been used in the field for years with the name "Unit Estimator". To estimate by the unit estimation method, the joint length of the block is used as an independent variables for each group of blocks which are grouped into 17 families.

In the experimental result, the neural network model significantly outperforms the multiple regression and the simple regression through origin.

DAS-LPP: Long-range Production Planning Expert System

The system aims to automating the generation of the shipbuilding production program which is the starting point of the production management and the important criteria for marketing activities (Yang & Choi 1994). To balance loads of production processes keeping the due dates of ship deliveries, we developed the expert system employing the beam search module which is connected with the linear programming solver MINOS. We used the linear programming for modeling the cumulative load curve (so called S-curve as in Figure 12) and employed the beam search procedure to select the best product mix cases reflecting the shipbuilding plants' status.

Implementation

For this project, we have used the frame-based expert systems tool UNIK('UNIfied Knowledge') which is developed by KAIST and upgraded from initial LISP version to C version. Since we own the source codes and can improve them, it was very adequate tool for this kind of project which requires flexible adjustment. UNIK has the features such as forward chaining (UNIK-FWD), backward chaining (UNIK-BWD), inductive learning (UNIK-Induce), rule generation from diagrammatic representation (UNIK-RuleGen), and neural network learning (UNIK-Neuro) and library of LISP-like functions(UNIK-Kernel). UNIK has the versions operating on UNIX , DOS, and MS-Windows. DAS systems were implemented on Sun SPARC station 10 series.

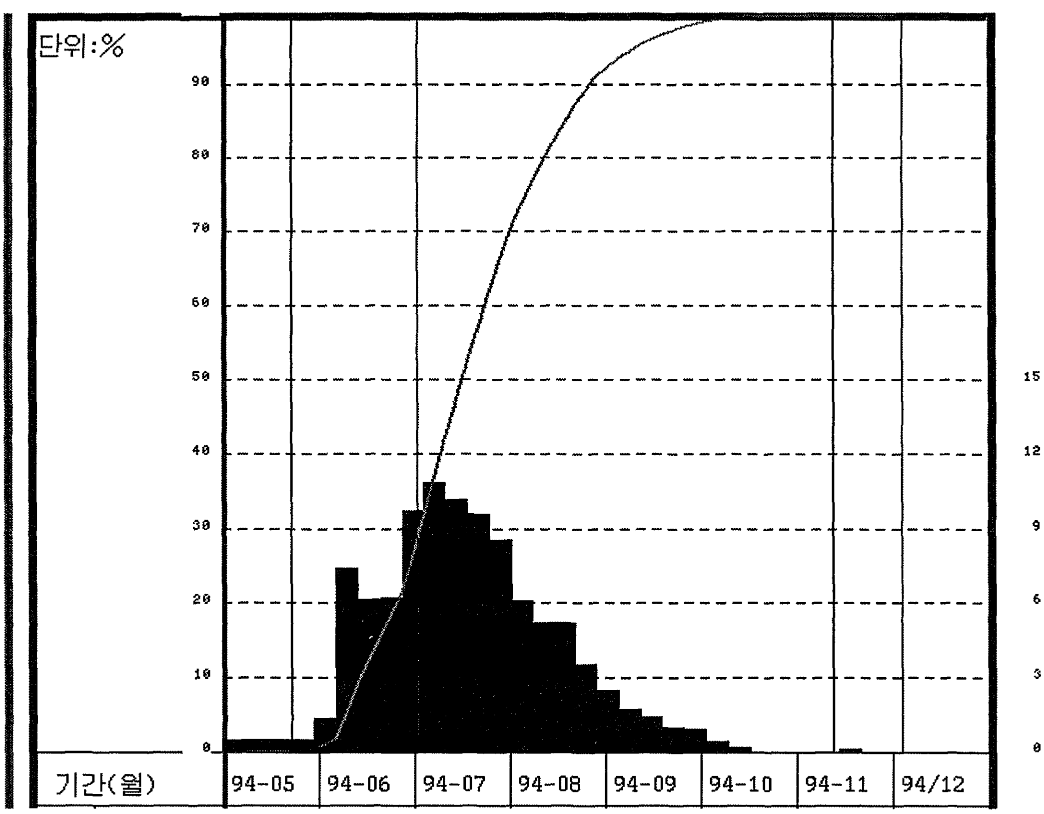

Figure 12. A Cumulative Load Curve (S-curve) used In DAS-LPP

	First year '91	Second year '92	Third year '93
DAS-ERECT	Research on Constraint Directed Graph Search	Spatial Scheduling in Pre-erection Shop, System Implementation, GUI Implementation	Experiments, GUI Development, System Upgrade & Maintenance
DAS-CURVE	Research on Spatial Scheduling	System Implementation GUI Implementation	
DAS-PANEL	Research on Line Balancing with Dynamic Cycle Time	Rule base & Inference Engine Construction, System & GUI Implementation	
DAS-MH	Tool (UNIK-Neuro) Development	Neural Network Modeling & System Development	Experiment & Model Selection
DAS-LPP		Research on CSP & LP modeling of the S-curve	Interactive Planning Prototype
Implementation Language & Tool	LISP Version Prototype	Operational System (C Conversion)	Installed in the field
Development Phase	Vision Revelation	Data Dependent Realization	Prospective Enhancement

Table 1. Development Process

Development Strategy and Deployment Process

For the large-scale and long-range project, a three-stage development process was devised (Lee 1993) as summarized in Table 1. In this phased approach, we considered not only user requirement but also data availability. The first stage was *vision revelation*. The KAIST team has persuaded that the DAS project is not a mere system development, but a theoretical research on the following issues:

1) Constraint-directed graph search for erection scheduling considering the work loads at the preceding indoor shops

2) Spatial scheduling

3) Line balancing with flexible process planning and dynamic cycle time

4) Processing time estimation using neural network

5) Interface and coordination among multiple expert systems and databases

These issues were explored during the first year, 1991, and prototypes of the DAS systems had been developed using the tool UNIK which had been implemented in LISP. These prototypes could successfully demonstrate the vision, although they could not provide the satisfactory speed.

The second stage was *data dependent realization*. Daewoo people were frustrated because they couldn't provide data necessary to run the prototype systems. However, it was a good checking point of identifying which data were available and which were not. This process required a close communication with the design division, because the initial prototype requires the information about the design specification from the CAD tools to automate the process planning and man-hour estimation. However, these data could not be provided until the current CAD system is upgraded to solid modeler which was not technically feasible at that time. The other data which could not be supported were the erection sequence dependent man-hour and the processing time requirements. Therefore, considering the data availability, the prototype systems were degraded. However, the management was delighted because the systems could run based on the currently available data although they were not ideally good. The systems in the second stage were developed using C++ version of UNIK to enhance operational speeds. The systems were installed in the fields in 1992 and took tests under the real world situation.

The third stage was *prospective enhancement*. During this stage, the systems developed in the second stage were incrementally enhanced as the prototyping approach does. However, a difference is that the enhancement was

	First year '91	Second year '92	Third year '93	Roles
University (KAIST)	9 persons	9 persons	6 persons	Research & Development Technology Transfer
	Research & Prototype Development, Knowledge Engineering	Inference Engine Development, Tool & System Conversion (LISP => C)	Advanced Implementation & Experiment	
People from company	3 persons	3 persons	5 persons	3 years residence in the university
	Specification, Transfer, Learning & Training	Rule Base & GUI Development	Maintenance and GUI Development	
Management team in the company		4 persons	4 persons	Coordination, On the Job Training
		Project Management	Data Support & Training	
Domain experts		11 persons	17 persons	In-house Training, Practice & Testing Support of Scheduling Masters
		Knowledge Transfer & Feedback Prototype Testing	Usage Learning & System Testing	

Table 2. The Number of Members in Each Group and Their Changing Roles

oriented toward the target set at the end of the first stage. The systems were gradually improved as the data became more available during 1993. This enhancement will continue as the necessary data become collected.

Project Work Group Organization

The project team consisted of the Daewoo people and the KAIST people (Table 2). It was good for the closer research and development to send the company's people to the university for overcoming the geographic distance (about 500 km) between the two sites. We observed that the roles of the project members evolved according to the development phases as shown in Table 2. It was smooth to work and communicate with the top management since the executives involved in the project were not changed until the end of the project.

Innovations at the site

The innovations by using the DAS systems can be summarized as follows.

- **Reengineering the planning and working processes**

  The aim of DAS was not automating the existing processes but innovating the old processes by AI technology. The followings are exemplar results of the innovations in processes during DAS project (Kim 1994).

  1) **Product mix case selection process**

     Before using DAS-LPP, the loads of the plants were calculated after product mix cases were selected. Now, we can select a good product mix cases considering the future loads of the plants.

  2) **Spatial scheduling process**

     Before using the spatial scheduling expert systems such as DAS-CURVE, the spatial allocation plan was made after the temporal allocation planning. Now, the integration of two scheduling processes saves much time and efforts, and enhanced the quality of schedule.

  3) **Expected man-hour calculation process**

     The previous inaccurate and heuristic 'unit method' is replaced by the neural network based estimation method.

- **Selection of the best schedule from simulation**

  The speed-up of the schedule generation time enabled the field scheduler to try multiple scheduling strategies and select the best schedule among the simulation results

- **Reduction of the scheduling effort from simultaneous and centered scheduling**

The networked scheduling expert systems in hierarchical architecture enabled the production management department to plan and control the plants totally reducing the negotiation efforts among plants.

- **Visual production management & control by graphical output**

  The graphical output screens generated from the systems like DAS-CURVE improved the planning and work productivity via clear communication between the management and workers.

- **Technology transfer from university to industry**

  DAS project is famous in Korea as a typical success case of the cooperation between university and industry. Now Daewoo has the ability of developing expert systems by herself.

Development Cost

The development costs for the three years were calculated at approximately $159,000 for the hardware and $675,000 for the research and developer of the both organizations. The total cost was approximately $834,000.

Estimate of Payoff

Though the revenue implication for the project is not easy to calculate, the company estimated it based on the expected contributing rate to the production productivity and planning productivity improvement rate. Since the estimated contribution rate of DAS to the yearly production productivity improvement(15%) is 30% and to the planning productivity improvement 50% respectively, the expected annual benefit by DAS project is about $4,000,000.

Maintenance

The systems are maintained by the programmers and knowledge engineers who were trained during this project. The schedulers can set the contingent scheduling strategy by setting the parameters for constraint base and rule base. For flexible system maintenance according to the change of plant layout and facilities, KAIST and Daewoo is currently conducting a research on the scheduling system generator as the next project of DAS, which is called DAS-II.

Acknowledgments

This research is funded by Daewoo Heavy Industries Ltd.(DHI) in Korea. We are particularly grateful to vice-president Dong Kyu Park, executive directors Wan Chul Suh, Yang Bin Cho, Hee Gyu Kim, department manager Se Jin Jang, section manager Gwang Joo Lee, and Mr. Sang Soon Im, Mr. Yong Chul Kim for their cooperation in knowledge acquisition. We also would like to thank members of DAS Project team at KAIST and Daewoo, who are Mr. Jae Wook Kim, Mr. Jeong Seung Lee, Mr. Jae Ho Chun, and Mr. Seok Jun Byun.

References

Hong, J. S.; Kim, E. Y.; and J. K. Lee. 1993. Assembly Line Scheduling Expert Systems in Shipbuilding: DAS-PANEL. In Proceedings of the Korean Expert Systems Society Conference (in Korean), 147-158. Seoul, Korea.

Kim, E. Y. 1994. CIM in Shipbuilding: A Case Study. In Proceedings of the Korean Expert Systems Society Conference (in Korean), 81-98. Busan, Korea.

Lee, J. K. and Kim, H. D. 1994. Man-hour Requirement Estimation for Assemblies Using Neural Networks, In Proceedings of '94 Japan/Korea Joint Conference on Expert Systems, 203-206. Tokyo, Japan.

Lee, J. K.; Choi, H. R.; Yang, O. R.; and Kim, H. D. 1994. Scheduling Shipbuilding Using a Constraint Directed Graph Search: DAS-ERECT, *Intelligent Systems in Accounting, Finance, and Management* 3: 111-125.

Lee, J. K.. 1993. Phased Development Strategy for Complex Expert Systems: A Shipbuilding Scheduling Case. In Proceedings of 93' Pan Pacific Conference on Information Systems, 255-257. Taiwan Republic of China.

Lee, J. K.; Suh, M. S.; and Fox, M. S. 1993. Contingencies for the Design of Scheduling Expert Systems, *Expert Systems with Applications* 6:219-230.

Lee, K. J. and Lee, J. K. 1992. Spatial Scheduling and its Application to Shipbuilding. In Proceedings of '92 The Second Pacific Rim Conference on Artificial Intelligence, 1183-1189. Seoul, Korea.

Lee, K. J.; Lee, J. K.; and Choi, S. Y. 1994. Spatial Scheduling Expert Systems for Shipbuilding. In Proceedings of the Second World Congress on Expert Systems, 243-249. Lisbon, Portugal.

Lee, K. J. and Lee, J. K. 1995. The Dominance and Preference of Search Space in Dynamic Spatial Layout.

Working Paper, KAIST-MIS-WP-9503-01, Dept. of Management Information Systems, KAIST.

Lee, K. J.; Lee, J. K.; and Choi, S. Y. 1995. A Spatial Scheduling System and its Application to Shipbuilding: DAS-CURVE. *Forthcoming in Robotics and Computer Integrated Manufacturing.*

Lozano-Perez, 1983. Spatial Planning: A Configuration Space Approach. *IEEE Transactions on Computers* 32(2):108-120.

Nilsson, N. J. 1980. *Principles of Artificial Intelligence*, Tioga Publishing Company.

O'Rourke, J.; Chien, C.; Olson, T.; and Naddor, D. 1982. A New Linear Algorithm for Intersecting Convex Polygons. *Computer Graphics and Image Processing* 19: 384-391.

Preparata, F. P., and Shamos, M. 1985. *Computational Geometry: An Introduction.* Reading, New York: Springer Verlag.

Yang, O. R. and Choi, S. Y. 1994. An Expert System with Optimization Model: DAS-LPP. In Proceedings of Korean Expert Systems Society Conference (in Korean), 241-251. Busan, Korea.

Zhu, D. and Latombe, J. C. 1991. Mechanization of Spatial Reasoning for Automatic Pipe Layout Design. *AI EDAM* 5(1): 1-20.