Proceedings of the 1992 Winter Simulation Conference
ed. J. J. Swain, D. Goldsman, R. C. Crain, and J. R. Wilson

SIMULATION OF MATERIAL DELIVERY SYSTEMS WITH DOLLY TRAINS

Vadivelu Jeyabalan
Norman C. Otto

Manufacturing Systems Department
Research Laboratory, Ford Motor Company
Suite 1100, 23400 Michigan Avenue
Dearborn, Michigan 48124

ABSTRACT

Well designed material delivery systems should support the production systems with timely delivery of material from its point of arrival to the plant to its point of use. This paper describes a simulation methodology to model and capture the behavior of systems which use dollies to deliver material. The models evolved over time by modeling several real world applications in a variety of manufacturing facilities. As compared to forklift systems, modeling dolly delivery is much more complex and requires several special features. Crate, dolly, and train entities merge to model loading at the dock and split to model dropoff at the aisles. Data structures are used to transfer and maintain information across these different entities. Decision logic must also be added to model dolly selection, train routing, and alternative actions when resources are not available. New control entities must be used to control loading dollies from the floor and periodic collection of empty dollies. All these techniques are described in the paper along with the model flow logic traced from dock unloading to aisle and line feed delivery to return and reloading of empties on the truck. Application of the models to a JIT implementation at an assembly plant and to the manufacturing facility design problem are also presented.

1 INTRODUCTION

Timely delivery of material from suppliers to the manufacturing, processing or assembly locations of a plant is an important aspect of the Just In Time (JIT) manufacturing concept. Streamlining that material flow requires adequate resource planning and proper design of plant material receiving and delivery systems. These systems consist of resources (docks, dollies, buffer areas etc.) and transporters (forklifts, tow tractors etc.) to support the delivery of material from its point of arrival at the plant to its point of use in the production system. In most cases, material is received at a plant in trucks or trains at docks and is consumed within a plant at various locations (line feed) along the machining, processing, assembly and manufacturing lines.

One of the primary goals of a well designed material delivery system is never to let those line feed locations starve for parts. Stopping a processing line for lack of input material is very costly because it disrupts production schedules and delays completion dates. Predicting the performance of these systems, however, is not a trivial task. Randomness (truck arrival times, downtime and repair time of transporters, production lines, and other resources) and the coupled interactions of the system design parameters (storage capacities, resource levels, plant layout, and operating rules) demand sophisticated analysis. Thus, discrete event simulation is used to model complex material delivery systems.

Material delivery within a plant can be either by forklift, by dolly, or by some combination of both. The choice depends upon the distances to be travelled and volume of material to be transported. Forklifts can carry and transport only one or two containers at a time and only for short distances (probably less than 500 feet). On the other hand, a large number of containers can be delivered long distances on dolly trains. High volume deliveries are possible because single dollies, each holding several containers, can be connected to form a train which can then be towed by a single tractor. Dolly train delivery systems are more complex to model than forklift systems, requiring additional data structures, and event constructs, and techniques to model system behavior. Simulation models of material delivery using forklifts are described in Jeyabalan and Otto (1991). As an extension, this paper discusses a simulation based approach to modeling material delivery by dolly trains that has evolved over time by modeling several real world applications. These experiences have been generalized in such a way that, by making only minor modifications and additions to the existing event routines, an entire class of material delivery systems can be modeled with the same basic program structure.

2 SYSTEM OVERVIEW

Figure 1 shows the typical layout of a plant and the three areas of importance to the material delivery systems; (1) Docks (2) Aisles and (3) Line Feeds. The dock area has the docks themselves as well as storage space for both full and empty crates. (Note that the terms crate, rack, load, and container are used synonymously throughout this paper.) Aisles are where the tow train drops full crate loaded dollies (F.C.L.D.) and picks up empty dollies (E.D.) and/or empty crate loaded dollies (E.C.L.D). Line feeds (LF) are material feed points located along the production lines. Each line feed has storage space for incoming full racks and the outgoing empty racks that are generated when all the parts in the full container have been fed to the production

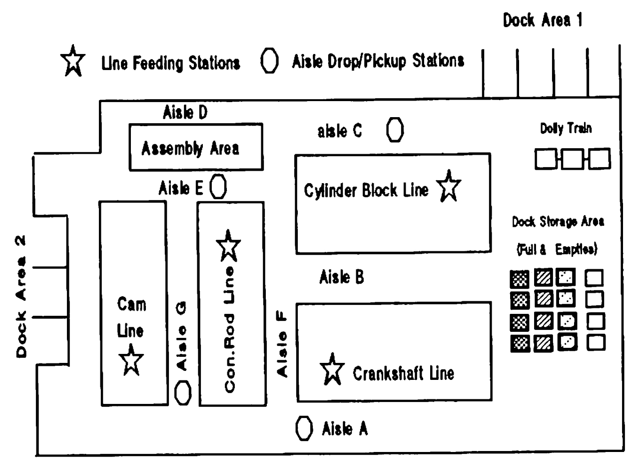

Figure 1. Material Delivery Areas

process. A single aisle can serve one or more line feed locations. A large plant usually has several groups (dock, aisles, and line feed) of these areas which do not share resources with the other groups. These smaller material delivery subsystems can be modeled and analyzed independently.

Figure 2 shows the flow of material through these three areas in a typical delivery subsystem. Trucks arrive at docks according to some pre-specified schedule (which can be varied), bringing incoming material in full crates (fulls). Each crate contains a specific part type and usually needs to be returned to the supplier when it becomes empty (empties). Fulls are unloaded from the truck by forklifts and loaded onto one of the dollies in a train.The train containing the F.C.L.D. is then pulled by a tow tractor to various destination aisles where the F.C.L.D. are dropped. The aisles to be visited by the train will depend upon what crates, or part types, are on the dollies of that train. The production lines are fed by forklifts which unload full crates from the F.C.L.D. dropped at aisles and bring them to the line feed location. Here, parts in full crates are consumed and empty crates are generated. LF forks transport those empties to aisles and place them on empty dollies. To utilize the train capacity during the return trip, any empty dollies, or dollies loaded with empty crates, available at that aisle are picked up by the train and returned to the dock.

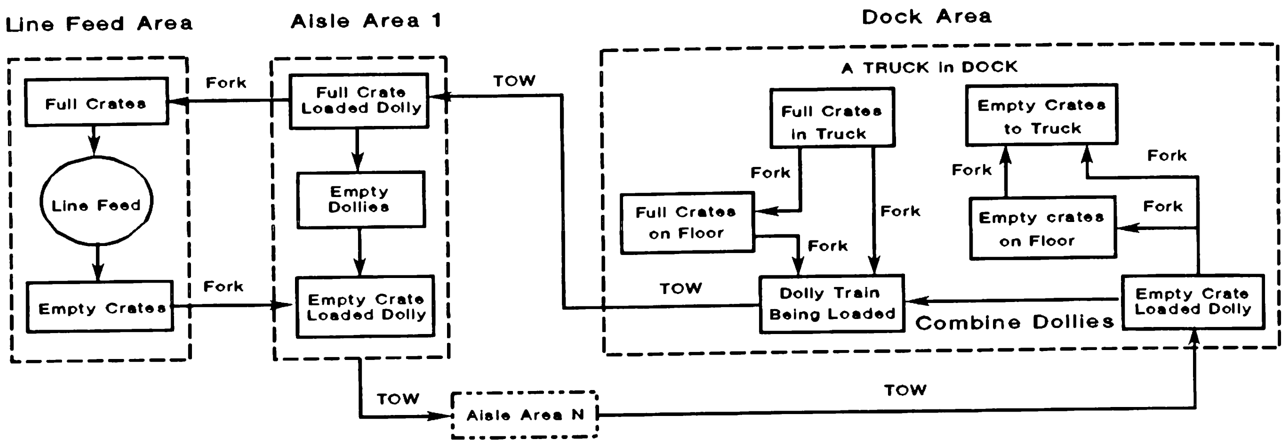

Figure 2. Material Flow

At the docks, empty racks are unloaded from returning dollies and stored on floor to await loading on a truck for return to the supplier. After returning to the dock, dollies and tows are available to transport more incoming material.

2.1 Complexities

In a previous paper, Jeyabalan and Otto (1991) described a relatively simple method for modeling material delivery systems using entities for the loads, resources for the docks and production lines, detached queues for storage locations, and stations as the transport locations. A control entity was used to monitor and control the truck loading and unloading processes. While the dolly delivery model incorporates all of these basic features, several additional complexities had to be accommodated.

2.1.1 Tow Train/Dolly/Load Relationships

The tow train/dolly/load combination differs in many ways from a loaded forktruck. The elements of this combination continually coalesce and split apart at various points in the system. To model these processes, methods are devised to merge and split entities using data structures (entity attributes and FORTRAN variables) to preserve all the information. In a merge, an entity transfers its data to the data structure of the entity with which it is merging and then disposes itself. The first entity can later be regenerated from the data structure of the merged entity. Figure 3 shows how this approach is applied to a dolly train material delivery system. When enough empty dollies are available to form a full train, these dollies are merged into a train entity, transferring the # of dollies data(arrows). Then, when a rack is loaded to the train, the part type, quantity, dolly position, and state (empty or full) information is transferred to the data structure of the train entity and the rack entity disposed. When the train reaches an aisle, the train data structure is used to generate the dolly entity being dropped off at that aisle and to transfer the information about the racks on that dolly entity. The dolly entity can now generate rack entities for transport to the line feed location. These same processes are repeated several different places in the system.

2.1.2 Decision Logic

A number of decision logic points, which were not needed in the forklift model, exist for dolly train delivery.

(1) Since dollies are a limited resource, it is likely that, at times, there may not be dollies available for truck unloading. When this occurs, a decision needs to be made and implemented in the model. One choice could be to continue unloading the truck but store the full racks near the docks for later delivery to aisles when empty dolly trains become

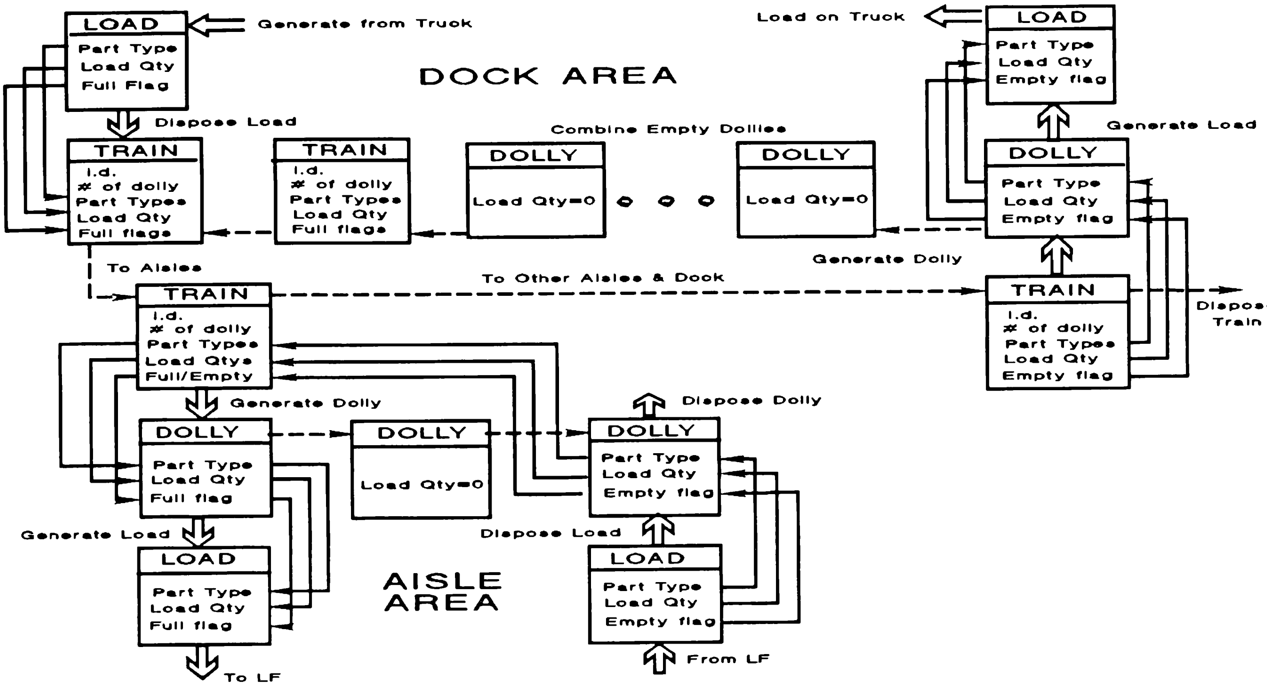

Figure 3. Data Flow

available. Another choice would be to stop the truck unloading process and make the truck wait for empty dollies.

(2) While unloading one or more trucks, space may be available on several trains that are being loaded at the dock area. Choices have to be made as to which train and which dolly to load. The model should permit the user to study the effects of different train and dolly selection procedures.

(3) Since several aisles usually have to be visited by a train, decisions have to be made regarding the route to follow for the delivery. If distance is of concern, the choice could be the next nearest station from the current station. If frequent detaching and attaching of dollies from the train is of concern, the route choice could be the destination aisle of the last dolly of the train.

2.1.3 Control Entities

Two new control entities were added, the first to retrieve full loads from dock storage when dolly space becomes available and initiate the dolly loading process and the second to make special trips to collect empty dollies from the aisles.

3 MODELING

The basic skeleton of the dolly delivery system model is written using the SIMAN/CINEMA simulation/animation language (V3.5). The model elements are basically the same as in the forklift model except there are more of them to account for the dollies and trains. A good deal of custom FORTRAN code is also used for the data structures and event constructs. These event constructs are used to trigger the merge and split processes, to execute decision logic, and to route the tow train. In addition, FORTRAN code is used to perform tasks such as accessing, from a flowing transaction, attributes of entities in detached queues, creating and disposing a large number of diverse entities, and implementing user options. In the following sections, we will explain the modeling procedures used for each of the major areas in the model.

3.1 Truck Arrival & Load Generation

The previous forklift delivery paper (Jeyabalan and Otto 1991) described how a truck generator control entity is used to generate truck entities and how the generated truck entity acquires a dock and, in turn, generates full rack entities as per truck composition and places them in a detached queue.

3.2 Truck Unloading

To unload the truck, a control entity (Truck_Unloader) is generated by duplicating the truck entity itself. This control entity derives most of its attributes from the truck (e.g. dock number of the waiting truck, number of current loads in truck, unloading priority etc.). Figure 4 shows the functions of truck_unloader control entity.

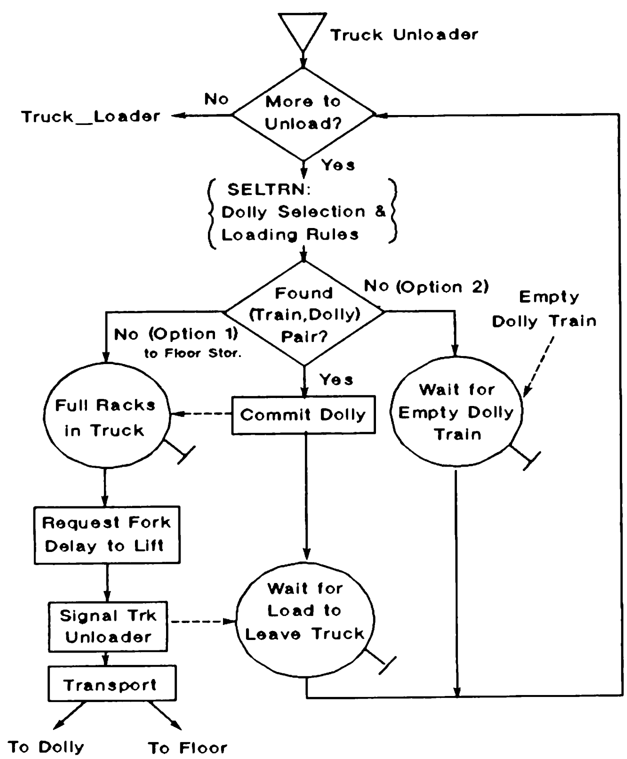

Figure 4. Truck Unloading

As a control entity, it can look at various detached queues in the system, their entities and the attributes of those entities. This entity starts the unloading process by executing the dolly selection procedure as described in the next section. At this point, the next action depends on the result of this selection process. If a dolly with the same part type as the rack being unloaded and with enough space for the load is found, the control entity commits that dolly space to this unit load so that no other load can claim that space. The truck_unloader then moves one load from the full crate detached queue to a fork request queue for transport to a dolly and waits until that load is removed from the truck. When leaving the truck, the load signals the control entity so that it can continue the truck unloading process. After the last load leaves the truck with fork, the truck_unloader control entity, knowing this fact, changes its name and role to become a truck_loader control entity whose function is to monitor the reloading of the truck with empty crates.

If, on the other hand, no dolly space is found, the truck_unloader can either, (1) wait at the dock for dolly space to become available or (2) send the load to a dock storage area. With the first option, the truck_unloader entity is put in a detached queue to await the arrival of an empty dolly train. With the second option, the procedure is the same as if dolly space was found except the rack entity is transported to floor storage instead of to a dolly. If dock storage space is available, the second option is usually preferred because it maintains the flow of incoming trucks.

3.2.1 Dolly Train Selection Algorithms

Selecting a dolly for the next load in the truck (SELTRN event in Figure 4) can be simple or complex depending upon the algorithm options chosen by the simulation user. At present, the following four dolly selection algorithms are implemented in the model.

Select

1. any dolly on ANY train (SELANY subroutine)
2. a dolly of a train DEDICATED to this dock (SELDCK subroutine)
3. a dolly of a train having already loaded racks going to same or neighboring aisles of current load (SELNN subroutine)

4. Combination of above algorithms in a specified order

In each case, the selected dolly must either be empty or have the same part type as the crate to be loaded and have enough space for the load.

3.3 Dolly Loading & Train Dispatching

When the load entity arrives at the selected dolly, it delays for the drop, frees the fork, reduces the number of committed loads, and adjusts the data structure of the train to reflect the addition of this load. If the train is now full and no other dollies are waiting for committed loads, the train is placed in a tow request queue for transport to an aisle. In addition, if a load is the last one from the truck, a partially loaded train will be sent to the aisle areas.

Full racks that are stored on the floor because dolly space was not available are loaded using the floor_to_dolly loader control entity. This entity is, in many ways, very similar to a truck_unloader entity. However, it is activated only when there are full racks in the dock floor storage, there is available dolly space, and no trucks are being unloaded. Once these conditions are meet, dolly loading from floor storage works the same as dolly loading directly from the truck.

3.4 Train Transport & Periodic Empty Dolly Collection

The NXTSTN FORTRAN event is called by all tow trains before they leave any drop/pickup station (aisle or dock area). This event determines what station to go to next and the travel time required to reach that next station. The next station chosen depends upon the type of train calling this event. There are two types of trains in our models (1) a regular drop/pickup train on a full delivery - empty pickup trip and (2) a periodic empty dolly pickup train. For the regular trip case, the routing decision depends upon what parts are in the full racks of the dollies and the aisles where those dollies are to be dropped. The data structure of the train entity contains the part type on each dolly which can be cross referenced to the drop aisle. In most of models, the next nearest aisle (among those aisles that need to be visited) is chosen.

Because of the system randomness, the generation of empty crate and dollies may not be synchronized with the delivery of full dollies. Thus, in order to have dollies available for use at the dock, it is necessary for a tow train to periodically pickup empty dollies from the aisles. To model this, the empty_dolly_collector control entity periodically leaves the dock station with a tow and a train i.d. but without any dollies. In this case, the station routing follows a predefined path. On returning to dock, the empty_dolly_collector entity frees the tow, drops any empty or empty crate loaded dollies in the appropriate storage queues and then waits for a pre-specified time before the next collection.

3.5 Aisle and Line Feed Areas

Figure 5 shows the activity in the aisle and line feed areas. When the train entity arrives at an aisle, it generates and sets the attributes of the full crate loaded dollies dropped off at that aisle. The train entity then goes on to pick up empty, or empty crate loaded, dollies from the E.D. or E.C.L.D. detached queues. The data structures of the train entity are adjusted to reflect the dollies dropped off and picked up. After delaying the train for unhook and hook up times, the updated train entity calls NXTSTN to determine the next destination and the travel time needed to reach that station.

Each F.C.L.D., before joining its detached queue, generates full rack loads (using the dolly attributes to set the load type) and sends each load to a fork request queue for transport to the line feed location. When the last load is removed by the LF fork, the dolly entity moves to the E.D. detached queue

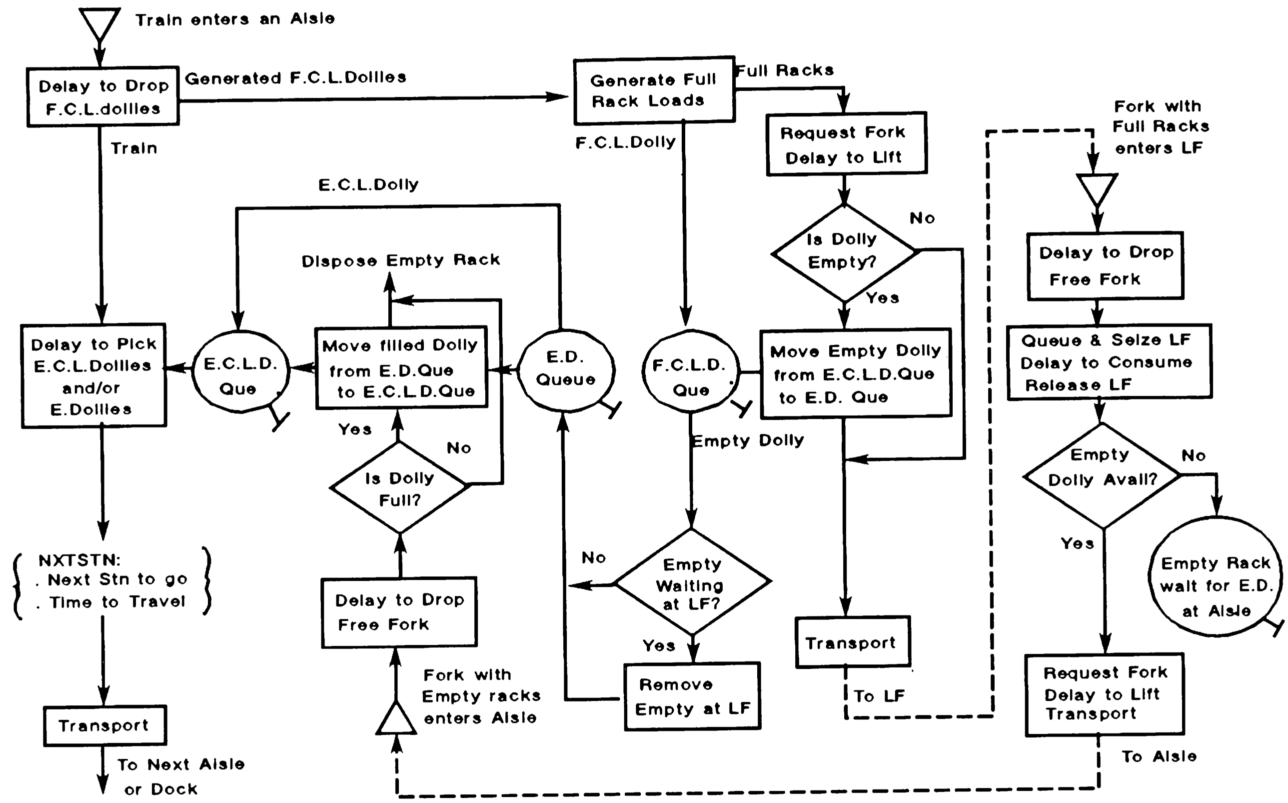

Figure 5. Aisle and Line Feed Areas

where it will be reloaded with empty crates generated at the line. Before entering this queue, the dolly entity checks for any empty crates waiting at the line for dolly space. The empty load entities brought from the linemerge with the dolly, transferring their attributes and then disposing themselves. If the dolly becomes full, it is moved out of the E.D. queue into a E.C.L.D. queue to await pickup.

As described in Jeyabalan and Otto (1991), the load entities at the line feed location seize the production resource for the time needed to use all the parts in the rack and then become an empty that requests transport back to the aisle. If no dolly space is available at the aisle, the empty waits in a buffer.

3.6 Dock Return and Truck Loading

The flow loop is completed when empty racks, brought back from the aisles, are reloaded on *the truck for return to the supplier (Figure 6).

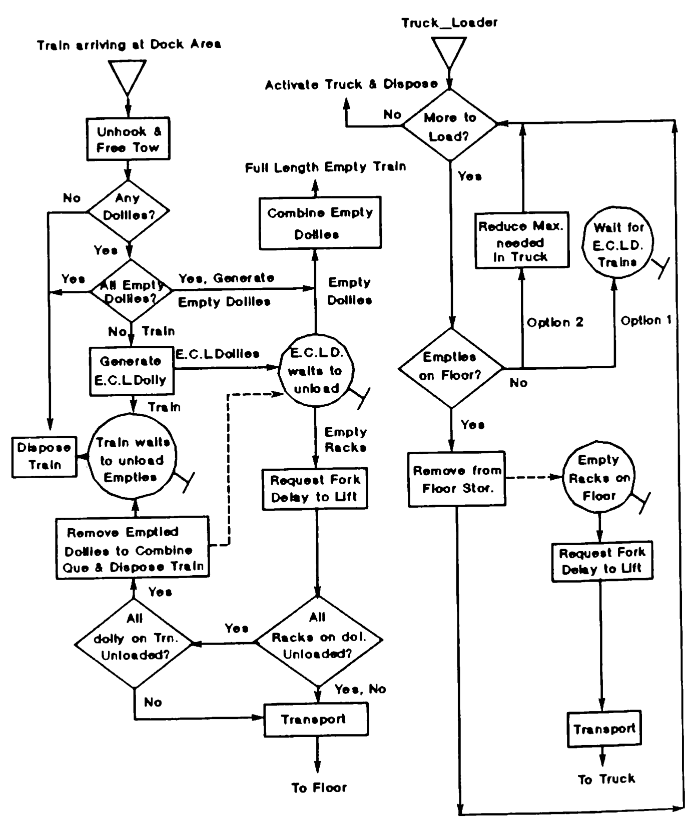

Figure 6. Truck Loading

When a tow brings a train entity back to the dock area from inside the plant, the train is unhooked, the tow freed, and the train unloading process begins. The train entity uses its data structure to generate dolly entitiesand transfer load information for each dolly on the train. If all the dollies are empty, the train entity is disposed and the dollies sent to the combine queue where they can form a new empty train entity. Otherwise, the dolly entities generate rack entities for each crate they contain and send these rack entities to a fork request queue to await transport to the dock floor storage area. When all loads have been removed, the dolly goes to the combine queue and the train entity is disposed. As mentioned in section 3.2, the loading of trucks with empty racks is controlled and monitored by the truck_loader entity. This entity is created when the truck_unload entity completes its task. The truck_loader first determines if the truck just unloaded has to be reloaded with empties. This decision, as well as the number of empties to return, is based on the return ratio concept discussed in Jeyabalan and Otto (1991). If empties are to be loaded, the truck_loader, knowing the type of racks required, searches the dock floor storage area, removes the appropriate empty load entity, and sends that entity to a forklift request queue for transport back to the truck. The empty racks arriving at truck are disposed after delaying for a drop and freeing the fork. This process is repeated until the truck has been reloaded with the required number of empties. If there are not enough empties available, the truck is usually sent back with a partial load to free up dock space. In either case, when the last empty is loaded on the truck, the truck_loader signals its associated truck entity out of waiting so it can release the dock for other waiting or future trucks.

4 APPLICATIONS

The primary application of these models is to find the minimum cost system needed to support production, where cost is directly linked to the number of resources (forks, tows, docks, dollies, and storage areas). Utilization statistics are used to monitor both performance and cost. The utilization of the production line resources will reveal whether the line is ever starved for parts. For the forks and docks, the number of resources are adjusted until a target utilization value is achieved (70 % for docks and 80% for transporters). For storage spaces, the average and maximum queue content allows these to be sized. In addition to the resource levels, a number of system parameters can be studied parametrically to determine their impact on performance. Included among these are plant layout (dock and aisle locations), operating rules (dock and dolly selection, train routing), truck composition and arrival pattern, and production line characteristics (rates, breakdown, etc.). In addition to the statistical output, supplementing the Siman/Fortran models with CINEMA animation allows visual study of tow train and forklift traffic congestion, dolly flow through the plant, and tow train routing.

Of the many applications addressed with these models, two are briefly discussed here. The first was a (JIT) implementation in an automotive assembly plant. Assembly plants have thousands of incoming parts that need to be delivered long distances to the line and, thus, efficient use of material delivery resources is imperative. In this particular study, the primary interest was in studying and comparing the performance of different dolly train selection algorithms. Among several algorithms studied, two are worth mention here. The first lets the truck unloader select ANY empty dolly at the dock, irrespective of the destination aisles of the other dollies on the train. In the second algorithm, the truck unloader looks for trains with dollies going to same or neighboring aisles in the plant. We want to use the model to quantify the benefit of minimizing the distance travelled by the tow train in a single trip.

As shown in Figure 7, the "nearest neighbor" dolly loading logic requires about two less tows to achieve the same level of utilization as the "any dolly" logic.

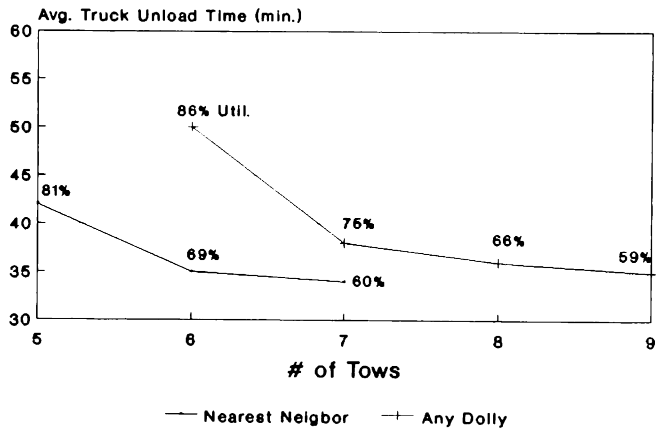

Figure 7. Performance of Dolly Train Selection Rules

Our second application was an existing engine manufacturing facility that had to be modified to produce a new engine type. In addition to receiving parts for existing lines, rough castings of the new engine type also had to be delivered to machining lines. Before physically moving existing lines to make room for new lines and before building new docks or changing the locations of existing docks, a layout analysis was done. Several possible alternate layouts of the plant were studied using our simulation model to obtain an optimal layout. The simulation study showed some important cost saving results:

1. A 12 spot dock planned for the new facility was underutilized. By routing additional trucks through those docks, not only was the dock utilizations increased, but a planned addition of a new 1 spot dock was also eliminated.

2. Since the part composition of most trucks was homogeneous (1 part type/truck), dolly train selection algorithms did not have a significant effect on either dolly or in truck unload/load times. Thus, the easy to implement "any dolly" rule can be used.

3. The empty dolly collection period could be once in 45 minutes to keep enough empty dolly trains available at dock area for loading full racks. The line feed forklifts were heavily underutilized and thus, a single forklift can be assigned to serve more than one LF location.

In summary, by using data structures, event routines and options for decision logic, material delivery systems using dolly trains can be modeled. By combining dolly delivery with existing forklift delivery models, we have a tool that can be used by material handling engineers to design and analyze any delivery system in a JIT environment.

ACKNOWLEDGEMENTS

The authors wish to thank Mr. John Handleman, Mr. Eric Jurs, Mr. Andrew Sowa and Mr. Paul Vij for their valuable contributions in making us understand the behavior of these material delivery systems.

REFERENCES

Jeyabalan, V. and Otto, N.C. Simulation Models of Material Delivery Systems. Proceedings of 1991 Winter Simulation Conference:356-364.

Pegden, D.C., Shannon, R.E. and Sadowski, R.P. 1990. Introduction to Simulation using SIMAN. McGraw-Hill, Inc.

AUTHOR BIOGRAPHIES

VADIVELU JEYABALAN is a principal computer applications engineer. His research interests are computer networking, manufacturing systems modeling and simulation.

NORMAN C. OTTO is a senior research engineer. His research interests include manufacturing systems analysis and computer aided engineering.