

MULTI-AGENT SIMULATION OF SAWMILL YARD OPERATIONS

Asif Rahman^{1,2}, Vald Stefan¹, Pascal Rebreyend¹, Siril Yella¹

¹Department of Computer Engineering, Dalarna University
Borlänge, Sweden

²Department of Information Technology, Örebro University
Örebro, Sweden

aus@du.se, h10vlast@du.se, prb@du.se, sye@du.se

ABSTRACT

This paper reports the findings of using multi-agent based simulation model to evaluate the sawmill yard operations within a large privately owned sawmill in Sweden, Bergkvist Insjön AB in the current case. Conventional working routines within sawmill yard threaten the overall efficiency and thereby limit the profit margin of sawmill. Deploying dynamic work routines within the sawmill yard is not readily feasible in real time, so discrete event simulation model has been investigated to be able to report optimal work order depending on the situations. Preliminary investigations indicate that the results achieved by simulation model are promising. It is expected that the results achieved in the current case will support Bergkvist-Insjön AB in making optimal decisions by deploying efficient work order in sawmill yard.

KEY WORDS

Logistics, Simulation, Multi-Agent, Timber Trucks, Log stackers and Sawmill yard

1. Introduction

The logistics optimisation techniques in industries are often concerned with maximising the profit, performance or minimising the cost, resources etc. In today's daily life most of the industries dealing with integrated supply chain logistics uses optimisation techniques in order to attain significant reduction in operational costs. The optimisation techniques applied in industries might be of different types. However, the use of agent based simulations for optimisation is not a new concept. Many simulations models are available in the market and are increasingly being used for optimization and decision making purposes. However, the currently available simulation packages are either commercial or open source, the open source packages does not provide any flexible environment for simulating the sawmill yard operations of Bergkvist Insjön, so we came up with our own simulation model to evaluate different scenarios of yard operations. Further, such scenarios will support Bergkvist Insjön decisions regarding scheduling and optimisation of their work order.

Despite being equipped with modern technology for handling operations, Bergkvist Insjön AB¹ is still facing several challenges thus threatening the overall efficiency and thereby limiting the profit margin of sawmill. A particular challenge concerns the efficient handling of resources (log stackers) from the time timber arrives and until it is fed to sawmill. The responsibilities of log stackers include unloading of logs from truck onto measurement buffer or on ground, emptying the storage bins (pockets) and piling up the logs in storage area and finally feeding the sawmill with logs for production. Given the huge production volume, it is obvious that the efficient handling of resources is a top priority. Unloading on the ground will cost more than double in handling because once unloaded on the ground the logs must be lifted again to be placed on the buffer whenever there is spare capacity available. Further, the sawmill owners can't keep timber trucks waiting in a queue as they are contracted to pay a waiting fee to drivers for keeping them waiting beyond certain time limit (30 minutes).

A simulation model representing the sawmill yard was developed to predict the performance of different yard operations under different what-if conditions. The what-if conditions which are included in simulation model are buffer size, working routines and unloading procedures. As a result of this, an accurate simulator describing the above activities will be available for testing different scenarios. The preliminary investigations indicate that the results achieved from this simulation model are promising and each scenario has its own advantages and disadvantages compared to other scenarios. Further, the comparison results suggested that the increase of the buffer size will be a good option to reduce the unloading on the ground but if we consider without any physical change then scenario 3 will be good alternative compared to other scenarios.

The rest of the paper is organized as follows. Section 2 Related Works, Section 3 Discrete-Event Simulations, Section 4 Multi-Agent System, Section 5 Experimental Design. Section 6 Results and Analysis and Finally Section 7 Conclusion and Future work.

¹ <http://bergkvist-insjon.se>

2. Related Work

The simulation-based model approaches to assess the operational routines within the domain and the impact of logistics is not a new concept. Many simulation models are increasingly being used for optimization and decision making purposes in every organization to optimize their resources costs and increase the efficiency, the number of commercially available simulation packages has also increased. Many newer simulation packages now offer object-oriented modelling architecture; therefore the simulation applications are more portable and scalable than before [1]. As enormous amount of literature available on simulation model, we focussed only on the Multi agent simulation models as the work done in this paper is limited only to agent based simulation models.

The problems in which agent technologies have been applied within transport logistics are planning, scheduling, management etc. Many researchers have focused on the field of multi agent systems. A number of significant advances have been made in both the design and implementation of agents. One of the successful approaches to agent based optimization is the concept of an asynchronous team (A Team), originally introduced by Talukdar *at el* [2]. Further, Davidsson *at el* [3] provides a survey on existing research based on agent-based approaches applied to transportation and traffic management. The focus of the authors was mainly on transportation due to its logistical perspective rather than traffic. In his survey author identified a number of positive aspects of the current agent-based approach to logistics. Agent technology seems to have contributed significantly to the advancement of state of the art.

However, agent-based approaches are often not evaluated properly. If the proper evaluation and comparisons with other technique can be done, then we can use agent technology to strategic decision-making with transportation logistics. The author further analyses the strengths and weakness of agent-based approaches and mathematical optimization techniques. According to the author's qualitative evaluation, agent-based approaches tend to be preferable when, the problem domain is large, probability of node or link failures is high, time scale of the domain is short, sensitive information that should be kept locally, and mathematical optimization techniques are preferable when, the cost of communication is high, the domain is monolithic in nature, quality of solution can be guaranteed, it is important that a system optimal or near optimal solution is found. Further, combining the approaches to make a hybrid approach can also be a promising way.

G. Vita and G. Janis [4], proposed an overview of Multi Agent System Architecture for solving problems in Transportation logistics domain using Agent Technologies. They conclude that, Multi-agent or agent-

based systems that mainly are used to solve typical problems in the transportations and logistics domain are decision support systems, logistics planning systems, and simulation and modelling. The different multi-agent architectures used by authors are holonic (autonomous self-reliant unit) agent technology and open system architecture. These architectures support both decision making and planning. Further, (Song *at el.*, 2001) [5] proposed an agent based logistics coordination and collaboration, where coordination agent coordinates each individual agent's goal searching activity by balancing different objectives and finds the "all-agreed" solution in a short time period. Nathan Nikotan *et al.*, (2011) have used multi agent system to schedule software projects. The main purpose of his approach was to exploit the parallel process of software scheduling and Monte Carlo simulation which is used to quantify the risk effect and uncertainty [6]. In another work, (Kyoungmin and Kyong ju, 2010) developed a multi-agent-based simulation to evaluate the traffic flow of construction equipment in construction site. The results were promising when compared with the conventional simulation system "SIGMA". It is expected that the results of this study will be useful in decision-making for the selection of an alternative process in order to complete a construction project with in the target period [7].

However, currently available simulation software doesn't provide flexible environment for simulating the sawmill yard operations of Bergkvist Insjön. To use that flexibility, communication and adaptability multi-agent based simulation has been proposed to simulate the yard operations. It is to be noted that, the work done by log stackers is acceptable but the problem lies in their performance while doing certain task which is ineffective and needless. To avoid those unnecessary decisions, improve the performance of log stackers and sawmill yard as a whole the present agent based simulation model has been developed and tested with different case scenarios.

3. Discrete Event Simulation

Discrete-event simulation is a well-known technique for modelling, simulating and analysing the systems where model describes a conceptual frame work of a system, simulation is performed on the computer implementation of that model and finally analysis is done to draw conclusions from that output to support in decision making process. The model, it can be characterized into different types such as stochastic or deterministic, static or dynamic and continuous or discrete. However, in this work the model is stochastic in nature and it can be described as an abstract representation of the system where it contains structural or mathematical relationship. This representation is made in terms of entities, their attributes, sets, processes, events, activities, and delays. In this modelling process, the "activities to the actors" approach was used. In this way, the existing activities in the sawmill yard will be identified first and those

activities will be modelled. Each individual actor will be created, communication is established, priorities will be assigned, and finally behaviour will be defined. By modelling all these activities of sawmill yard we can identify the problems in the current working routines and possible solutions to solve these type of problems.

4. Multi Agent Based Simulation Model

An intelligent agent is an entity that is capable of flexible autonomous action in some environment, where flexible means reactive, proactive and social. The reactive agents are those who can perceive their environment and respond in a timely fashion to the changes that occur in it, in order to satisfy their design objectives. The pro-active agents are those who can exhibit goal directed behaviour by taking the initiative in order to satisfy their design objectives. The social agents are those who are capable of interacting with other agents, in order to satisfy their design objectives. In this multi-agent based simulation model agents will interact with each other and their environment. The agent receives information from other agent and they have some internal rules or objectives that represent the decision process and how they should respond to each other's request. However, the choice of modelling the agent's behaviour will depend upon the application context, so the agent's behaviour in this simulation model varies in each scenario.

Due to the various characteristics of the agents and the ability to mimic the real world situations or scenarios, this type of modelling approach appears to be more suitable than Discrete Event System Simulations for modelling the sawmill yard operations. The agents, responsibilities, behaviour and mode of communications have been briefly described.

4.1 Sawmill Agent

The sawmill agent simulates the process of production house 'sawmill' where it has to process all the logs which are measured by another agent called measuring station agent. The sawmill agent has a buffer with a capacity of 300 logs and its maximum processing speed is 20 logs/minute. The sawmill agent sends a request message to resources (Log stacker) agents to fill the buffer with any random class of logs. After receiving the request, log stacker agents will communicate between each other before responding to the request. If priorities are assigned then the agent who is responsible for refilling the buffer will proceed further. This sawmill agent also manages the storage area of sorted timber. The logs which are emptied from pockets are piled up in a storage area and these logs will be used for production.

4.2 Measurement Station Agent

Trucks loaded with timber arrive at the sawmill yard at Bergkvist where they are unloaded by a log stackers.

Unloaded timber passes through a measuring station in which a human inspector checks for the quality of the timber. After this the timber passes through a 3-D scanner that will determine the type and dimension of the wood. Depending upon the type and dimensions, timber gets ejected from a conveyor belt into appropriate log storage bins (pockets) for further usage. Depending on the production requirements, timber from specific storage bins is emptied and fed to sawmill with the aid of log stackers. In this simulation model, the dimensions, measuring and unloading time were predefined on the basis of historical data analysis and on-site monitoring.

The measurement station (MS) has a buffer with two carriages and each carriage has a capacity of one truck logs. The sorting time is 10 minutes and unloading time is 5 minutes for one truck of logs. The sorting time is taken as an average of minimum and maximum number of logs arrived in each truck over couple of years. And unloading time is taken from on-site monitoring. However, the log stacker will take 10 minutes to fill the buffer with two carriage and 15 minutes to fill the buffer with three carriages (Scenario 1). When the MS starts the process of measuring logs, the agent will process the logs from the first carriage and then shifts the second carriage logs onto the first carriage. Then, the second carriage will be refilled with new logs, this process will continue until the end of shifts.

Logs Type	Total logs	Class	Number of logs	Class Ratio
Pine	1680185	F13	25257	0.015
		F14	58547	0.035
		F14 L 41	15629	0.009
	
		F32	5844	0.003
		F35	2171	0.001
		F37	863	0.001
Spruce	1219249	G13	28734	0.021
		G14	40724	0.032
		G15 L 41	49757	0.038
		G15 L 51	17852	0.014
	
		G34 L 56	2641	0.002
		G34 L 61	2197	0.002
		G37	1606	0.001

Figure 1: Percentage ratios of Spruce and Pine classes

Based on the historical data, the number of logs arrived in each class has been calculated and percentage ratios were assigned. The number of logs unloaded on the measurement buffer either pine or spruce will be multiplied by the above ratios and resulted logs will be stored in individual pockets of each class. However, more focus is laid on the work order of log stackers and the exact analysis of log dimensions or classes has been left for the future work.

4.3 Pocket Agent

The pockets are filled by measuring station agent with respect to percentage ratios as mentioned above. For emptying the pockets, measurement station sends a message to the pocket control agent at the end of each sorting cycle (one truck/carriage). This message contains the type of last log which has been sorted. If the last sorted log was pine, then the pocket control agent checks all the pockets between F13 and F37 (Pine classes) for every 90 seconds. If any of the pocket is full (>80), the pocket control agent sends a message to the log stacker agents to empty the pocket. After 90 seconds, the pocket control starts a new checking cycle and it continuous until new message is received by pocket agent. Further, if both log stackers are busy, the one who finishes the task first will be called and if priorities are assigned then the prioritized log stacker will empty the pocket.

It is worth mentioning that, at the time of emptying the individual pockets the sum of each pocket will have decimal value because of ratios which is different compared to reality. To avoid problems with decimal values we empty only the integer value logs from the pockets, so that the further processing will be in integers and decimal values will stay in pockets throughout simulation. (E.g. After sorting few trucks the sum of each pocket will be around 40.5, 84.5, 75.9 etc., we empty the pockets only when they cross 80 logs, in that case we take out 80 and keep the reaming 4.5 in the pocket).

4.4 Log Stacker Agents

The log stacker agents are one of the most important resources of this simulation study. The role of each log stackers varies according to priorities and scenarios. However, the responsibilities of log stackers include unloading of logs from truck onto measurement buffer or on ground, emptying the storage bins (pockets) and piling up the logs in storage area and finally feeding the sawmill with logs for production. The time spent in travelling by log stackers for each operation is given below.

Table 1
Log stackers' timings in seconds

Travelling	Sawmill	Truck	Pockets F	Pockets G
Sawmill	0	90	60	90
Truck	90	0	60	120
Pockets F	60	60	0	90
Pockets G	90	120	90	0

On site monitoring reveal that, each log grabbing operation is determined to last 30 seconds. The truck unloading process takes 300 seconds due to the fact that every timber delivering truck has three carriages and each carriage contains some logs. The log stacker needs 3 (maximum) grabs and movements between the truck and

measurement station buffer/ground to completely unload the truck. This means that, it will take 5 minutes to unload one truck on top of the measurement station buffer or on the ground. If truck is unloaded on the ground, then the log stacker's load time will be extended to 10 minutes. Five minutes for unloading on the ground and another 5 minutes to load it back on the measurement buffer whenever it's free. .

The course of action for each log stacker will be different and they have their own priorities. For a log stacker to do certain task it will take certain amount of time. The time for each task varies according to the position of log stackers and the task they are assigned. For example, if one truck is to be unloaded and the required log stacker is placed at pocket G, then the total time taken by the log stacker to finish the task will be around 420 seconds, described as follows:

- Traveling from the pocket G to the truck will last 120 seconds;
- Unloading of the truck will last 300 seconds (There are several grabs and time spent between the truck and the measurement station);

The sawmill supply task is accomplished by three different movements between the sawmill buffer and the specified storage area of sorted logs. However, this time also varies according to log stacker's positions and the scenario priorities.

- Grabbing of the F13 logs will last 30 seconds;
- Traveling from the storage area (pocket F13) to the sawmill will last 60 seconds; and continue until buffer is full.

4.5 Dispatcher Agent

Once the connection is established between database and simulation model, the dispatcher agent will read the predefined database table and launch the truck agents one after another. The time difference between each launch will vary as it is based on real data set (truck arrivals). After launching first truck the agent will check the time difference between first and second truck, then waiting for that many minutes/seconds the agent will launch the second truck and similarly compare second with the third and it goes on until all truck are launched. After launching the trucks the dispatcher agent will send a request message to unload the newly launched truck. However, the response to that request will depend on the current scenarios and priorities of log stackers.

4.6 Truck Agent

As the truck arrives, it will wait until who responds to the request message sent at the time of launching. However, the response of log stackers varies according to priorities and case scenarios. After receiving the requested message

and completion of its unloading procedures the truck agent will leave the simulation model. Each truck can only transport either pine or spruce and it is identified by its unique identity code mention in database.

5. Experimental Design of Multi Agent Simulation Model

There are few commercially available packages which can be used to simulate the model but the current model is proof of concept so we didn't use any commercial packages, may be in future. However, in this work, JADE (Java Agent Development Environment) is used as it is the closest solution available to develop a multi-agent system. JADE is a software framework fully implemented in Java language [8]. It simplifies the implementation of multi-agent systems through a middle-ware that complies with the FIPA specifications and through a set of graphical tools that supports the debugging and deployment phases. The Foundation for Intelligent Physical Agents (FIPA) defines and controls the above mentioned specifications, under a collection of standards, which are intended to promote the interoperation of the agents and the services that they offer. For the developing this simulation in JADE, we used Net Beans Integrated Development Environment (IDE) 7.0, available for free at [9]. The database model component was developed and maintained in MySQL software. MySQL is a relational database management system (RDBMS) which was chosen because of its easy implementation with java environment. MySQL development project has made its source code available under the terms of the General Public License (GNU), as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single non-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation [10]. MySQL database has become the world's most popular open source database because of its high performance, high reliability and ease of use. In this work, three different simulation scenarios were developed and tested in a dynamic environment of sawmill yard.

5.1 Data

The design in this methodology starts with the data acquisition process, where a raw data is acquired from Bergkvist Insjön AB and stored in the database. The main objective here is to test the simulation on real data set. The time of truck arrivals and the type of load they are carrying are exactly same as real data except the distribution of log class and numbers of logs in each truck were according to statistical analysis and random numbers generation respectively. To test the simulation, the busiest day in a year has been selected and all the scenarios were tested with the same data set, to test the credibility of this simulation model.

5.2 Conventional Model:

In this model, the simulation will simulate the existing working routines of the Bergkvist Insjön AB. Here, the resources used for handling operations of sawmill yard will proceed with the requests coming from truck agents, measurement station agent, pocket control agent and sawmill agent. When a request (message) arrives, both the log stacker agents will communicate with each other and decide who will respond to that request. Whenever a message arrives it is the responsibility of a dispatcher agent to check whether the log stacker is busy or not by considering the busy variable. If one log stacker is busy then forward the request to another log stacker and vice versa. If both are busy then wait until who finishes their respective tasks first and respond to the request. Further, in this model the unloading capacity on measurement station buffer is only two trucks, if there is a queue the trucks have to wait until the buffer is empty or else they will be unloaded on the ground.

5.3 Test Case scenario 1:

In this case, the measurement station buffer was increased with one additional carriage (C3) as seen in the figure 2. Initially in the conventional model we have only two carriages (C1 and C2). Now, in this case the log stackers can unload three trucks on the measurement buffer. If we assume that the measurement station is running, the only gain by increasing the buffer will be that for every 25 minutes of log unloading on the buffer the log stacker will gain 20 minutes break from the measurement station. For 20 minutes the log stackers can do some other work if there is any, if not then stand still. Because standing still is lot better compared to running without any load. Further, in this model there is a possibility of unloading trucks on the ground if there is a queue. The 20 minutes break can be described as, to fill the three carriages it will take 15 minutes and in same time measurement station will process one and half carriage logs. Further, two more truck can be unloaded as the buffer keeps on emptying. Finally, giving log stacker 25 minutes break, 20 minutes can be used in some other work and 5 minutes to fill the buffer again.

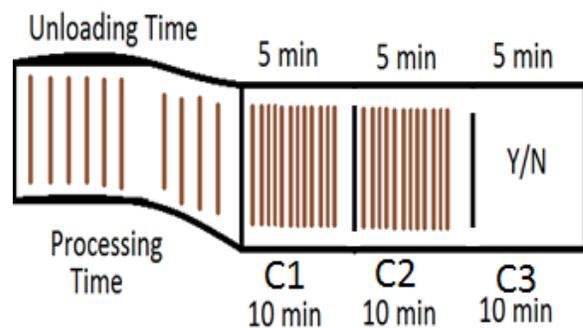


Figure 2: Measurement Station Buffer

5.4 Test Case scenario 2:

In this case, the scenario is similar to case 1 where the extra measurement buffer carriage has been used. However, the only difference here is that, there is no unloading on the ground. No matter what, the trucks have to wait until their turn comes. The lack of unloading on ground significantly increases the waiting time of arrived trucks. Further, the Bergkvist Insjön AB is obliged to pay a waiting fee for truck drives. In this case, the responsibilities of log stackers are similar to the conventional model and case 1.

The responsibilities of Log stacker one with respect to their priority is as follows

1. Unload the arrived trucks on buffer or on ground
2. Fill the measurement buffer either from trucks or from ground
3. Unload the storage bins
4. Supply the logs to sawmill for production

The responsibilities of log stacker two with respect to their priority is as follows

1. Supply the logs to sawmill for production
2. Unload the storage bins
3. Unload the arrived trucks on buffer or on ground
4. Fill the measurement buffer either from trucks or from ground

5.5 Test Case scenario 3:

In this case, the responsibilities of log stackers were distributed. The log stacker 1 is responsible for unloading the arrived trucks and filling the measurement station buffer whereas the log stacker 2 is responsible for filling the sawmill buffer with the required logs and emptying the storage bins. However, the buffer size in this case is similar to the conventional model and there is unloading on the ground as well. Which means the only difference between the model and case 3 will be the workload of log stackers. Here the log stackers will not share the responsibilities, they will only respond to the tasks assigned to them, whereas in model the log stackers were used to share the responsibilities.

6. Results and Analysis

The conventional approach and 3 case scenarios were evaluated and compared with each other after several trails. The results were different each time due to the random number of logs in each truck. That randomness effect the pockets storage and those pockets engage the log stackers more often which makes the results non-deterministic. The evaluation and comparison process has been further divided into different sections, which are as follows.

6.1 Measurement Process Speed

The MS performance has been evaluated with respect to number of trucks unloaded on the buffer or on the ground in percentages. The performance of measurement station isn't dependent on the buffer size, the increase in buffer can give extra time for log stackers but measurement station speed remains constant unless some stoppages in between. However, the results achieved from simulation model suggested that more trucks were unloaded on the ground, if those unloading can be avoided the performance will be much better.

Table 2
Measurement Stations Performance

Case	No of Trucks	Buffer (%)	Ground (%)
Model	123	43	57
Case 1	123	59	41
Case 2	123	100	0
Case 3	123	35	65

6.2 Sawmill Process Speed

The sawmill performance is evaluated with respect to the number of logs processed and number logs supplied by each logs stacker. Sawmill processing is the final step in the sawmill yard operations, therefore number of logs processed by sawmill measures the overall performance of sawmill. As stated before, sawmill processing speed is 20 logs/minute and 1200 logs per hour. Therefore the sawmill performances depends directly on the speed at which the buffer is filled. If it isn't filled regularly then it will stop, until the buffer is filled. The second and third scenarios have the most significant improvement in the sawmill performances. Due to the fact that, in case 2 the log stackers will not unload any waiting truck, just fill the buffer and do some other work. Whereas in case 3 the log stacker 2 is completely dedicated to feed sawmill and unload the pockets.

Table 3
Numbers of log processed in sawmill

	Model	Case 1	Case 2	Case 3
Overall Logs	22262	22537	26056	26643
Log Stacker 1	13328	14581	8963	-
Log Stacker 2	8934	7956	17093	26643

6.3 Truck waiting Times

The trucks waiting time is calculated from the time they arrive at the sawmill yard and until they are unloaded by the log stackers. The waiting time for each truck can vary depending on the scenarios used. However, it is to be noted that sawmill owners have to pay a waiting fee to truck driver for keeping them waiting beyond certain time limit. The results suggested that in test case scenario 2 has

an extraordinary increase on the waiting time. This is expected since there is no unloading on the ground which increases the waiting time.

Table 4
Truck waiting Times in Minutes

Minutes	Model	Case 1	Case 2	Case 3
Average	36.3	29.9	176	12.7
Maximum	123	99	412	48

6.4 Log Stacker Time Distribution

To increase the overall performance of sawmill yard operations the log stackers have to be managed very efficiently. Such that, run and load time can be minimized for both the log stackers. The operating time of the log stacker was divided into three parts:

Load time: Time spent by a log stacker on a specific task (with load).

Run time: Time spent by log stacker moving between different locations in the sawmill yard (without any load).

Idle time: Time spent by log stacker waiting for a task

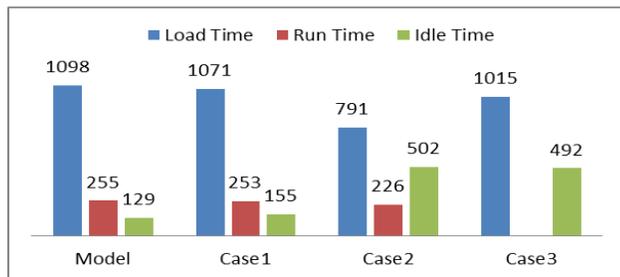


Figure 3: Log stacker 1 time distribution in minutes



Figure 4: Log stacker 2 time distribution in minutes

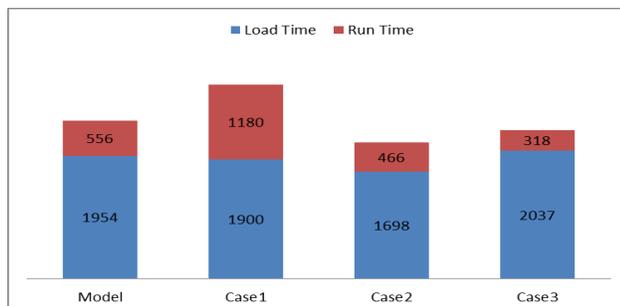


Figure 5: Working time of both the log stackers in minutes

7. Conclusion and Future work

To improve the performance and optimize their resources a multi-agent simulation model has been proposed and tested with different case scenarios. The advantage of using multi-agent based simulation model compared to other classical simulations is to predict the system performance and its variations. In this model, individual agents are programmed to follow simple behavioural rules or even interact with other agents in their environment in order to achieve their common goals. Further, with this type of models we can assess their individual performance and their behaviours with respect to various what-if conditions or routines.

The physical changes in the sawmill yard does effect the performance of yard operations. However, by increasing the buffer size we decrease the load time of log stacker and increased the idle time, but the waiting time of trucks causes more problems to Bergkvist Insjön than the idle time of log stacker. The truck waiting time has a hidden cost which cannot be seen in simulation model. The buffer size increase is a good option but it is quite difficult to implement it in real time. Whereas booking the arrival time is another good option to avoid the queuing of trucks but it takes time for drives to understand the flexibility involved in the approach [11]. However, without any physical changes in the sawmill yard the only way we can improve the performance is by dividing the work order between two log stackers (case 3) and try to avoid unloading on the ground as much as possible. Because, that's the only disadvantage in case 3, if it's avoided then it will be a good option for implementing it in real time. Where log stacker responsible for sawmill and pockets should only concentrate on those things whereas the log stacker responsible for unloading trucks should also feed sawmill and unload pockets if there isn't any trucks waiting in a queue to get unloaded. As mention earlier for every 15 minutes of unload on buffer the log stacker will get 10 minutes free which can be utilised in proper planning to do some other work and 5 minutes to fill the buffer with logs so that measuring station doesn't stops.

Regarding the validation of the model, the experiment is a proof of concept to demonstrate the possible scenarios to optimize the working routines. However, there are many things to be considered before validating the results. The processing time for all the logs in sawmill were constant irrespective of their dimensions and processing time near measurement station were also constant throughout simulation. Which is not the case in real time, however, there are only two scenarios (conventional and case 3) in the model which can be validated in real time but the results cannot be compared as the log distribution is different compared to reality and breakdowns and stoppages were not included as often these things happens without any prior knowledge and it will affect the behaviour of measurement station and log stackers. These drawbacks will be addressed in the future work, where the

processing speed of each class and distribution of logs in each truck will be similar to reality and dynamic pockets will be used to ease off the log stackers.

References

[1] E.J. Chen, Y.M. Lee, P.L. Selikson, "A Simulation study on logistic activities in a chemical plant" 2002.
 [2] S. Talukdar, L. Baeretzen, A. Gove and P. de Souza, "Asynchronous teams: Cooperation schemes for autonomous agents" 1998.
 [3] P. Davidsson, L. Henesey, L. Ramstedt, J. Törnquist and F. Wernstedt, "Agent-Based Approaches to Transport Logistics" 2005
 [4] G. Vita, G. Janis, "Technologies and Multi-Agent System Architecture for Transportation and Logistics Support: An Overview
 [5] Z. Song at el., "Agent-based Logistics Coordination and Collaboration" 2001

[6] N. Nikotan, J. Hab & M. Behesthi,"Software Project Scheduling Using a Multi-Agent System", Proceedings of 2011 Eighth International Conference on Information Technology: New Generations.
 [7] K. Kyoungmin and K. Kyong Ju, "Multi-agent-based simulation system for construction operations with congested flow", *Journal of Automation in Construction* 19 (2010) 867-874
 [8] Java Agent Development Framework, [Online], Available from <http://jade.tilab.com/> , [Accessed 21 December 2011]
 [9] Net beans, [Online], Available from: <http://netbeans.org>, [Accessed 21 December 2011]
 [10] MySQL [online] Available from: <http://en.wikipedia.org/wiki/MySQL>, [Accessed 21 December 2011]
 [11] Asif R, Siril Y, Mark D, "Image processing technique to count the number of timber logs in a truck", Proceedings of Signal and Image Processing 2011 Dallas, USA

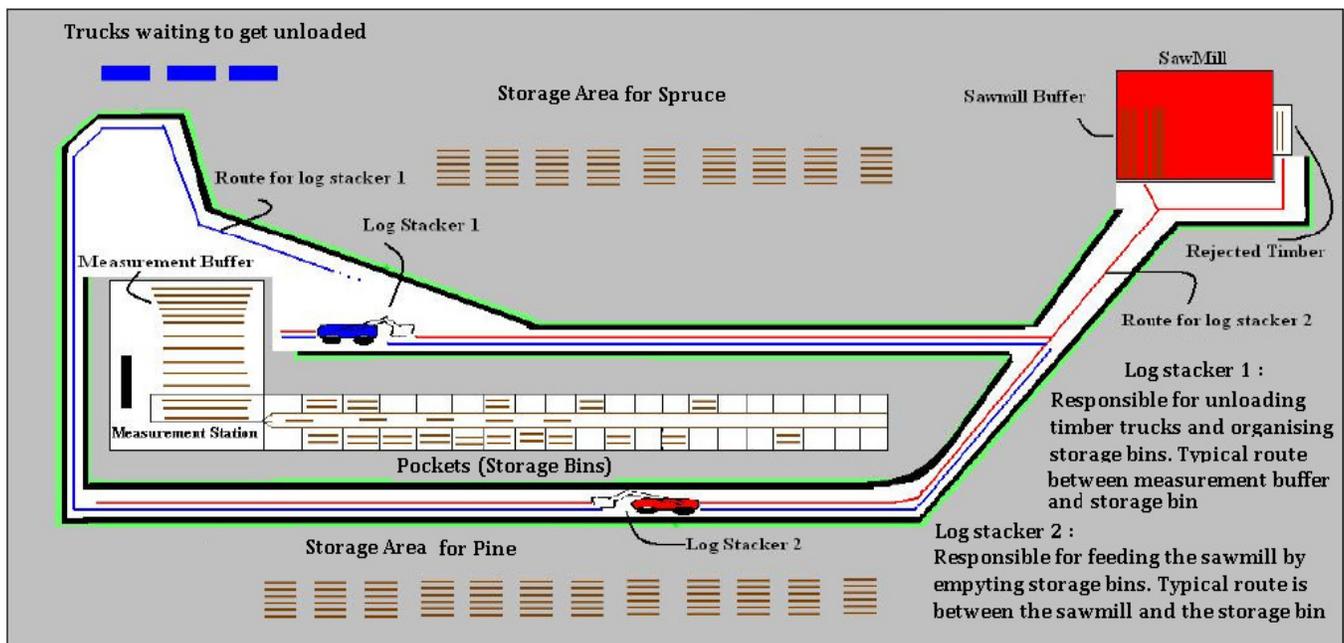


Figure 6: Sawmill yard