5th International Industrial simulation Conference ISC'2007 June 11-13 Delft, The Netherlands
Eurosis-ETI. ISBN 978-77381-34-2

BULK TERMINAL MODELLING AND SIMULATION

Jaap A. Ottjes, Gabriel Lodewijks and Dingena L. Schott
Delft University of Technology
Faculty of Mechanical, Maritime and Materials Engineering
Mekelweg 2, 2628 CD Delft, The Netherlands
j.a.ottjes@tudelft.nl; g.lodewijks@tudelft.nl; d.l.schott@tudelft.nl

KEY WORDS

Transportation, Bulk Terminal, Process Interaction Modeling, Operational Control

ABSTRACT

A modeling approach for operational control and evaluation of large export bulk terminals is presented. The model works at individual equipment level including the equipment break down behavior. As key performance indicator the delays of the bulk carriers is taken. The results of an example case are shown determining the maximum capacity of a terminal for different stock pile configurations. A view is given on future research.

INTRODUCTION

Deep sea export bulk terminals are interfaces between mines and bulk carriers for deep sea transport. The heavy equipment and the substantial area needed for a bulk terminal require large investments and the operational costs are considerable. The demand for raw materials such as iron ore is expected to increase strongly in the next decades. As a consequence new terminals will have to be built and existing terminals are forced to expand.

The operational control of a terminal and the terminal design (layout and equipment configuration) are strongly related. It is anticipated that it is possible to improve operational control considerably by using modern techniques especially by taking into account dynamic aspects.

Minerals like iron ore or coal are extracted from inland mines at a large scale and transported, usually by train, to a deep sea export terminal for further transportation to import oriented bulk terminals. The bulk material is temporary stored on a stock yard and later reclaimed and loaded into bulk carriers. The key performance indicator of such a system is the waiting time of bulk carriers (Jagerman, 2003). The waiting times depend on the capacity and the reliability of the total chain of equipment components of the terminal. The reliability is a combination of reliability of all individual equipment: train unloaders, conveyors, stacker-reclaimers and ship loaders. The reliability of the individual components can be expressed as the availability and the distributions of down- and up-times of the component. An example of simulation of a specific terminal is described in Ottjes and Lodewijks (2004).

In order to determine the terminal performance as a function of many variables and to test alternative operational control methods, a simulation model has been developed. In this paper the modeling approach will be discussed and an example of the use of the model will be shown.

TERMINAL EQUIPMENT AND OPERATION

In this work we concentrate on export oriented dry bulk terminals. Because the design process of both import oriented and export oriented bulk terminals are the same and the layout design considerations are similar, the modeling approach discussed is applicable to both terminal types.

We consider a terminal that is fed by trains. Several material grades may be involved for example originating from several mines. Figure 1 shows a simple terminal lay out with equipment.

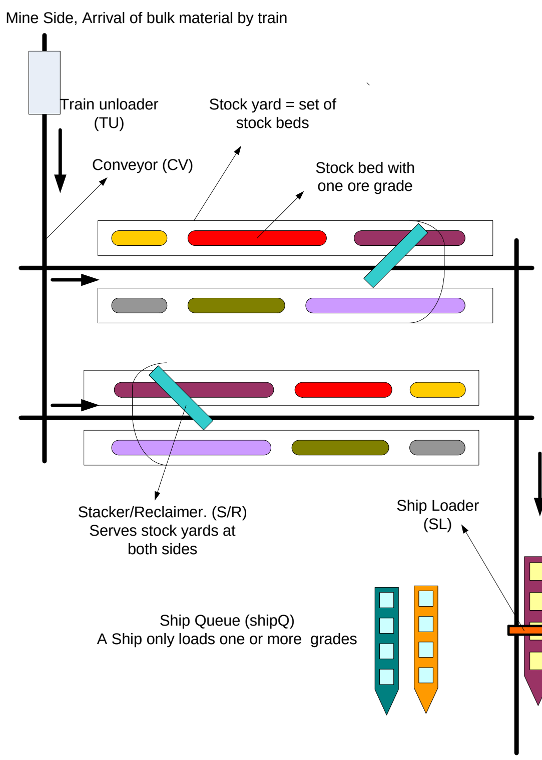

Figure 1: Terminal with Stock Yard and Stock Bed Lay Out

Train trucks are unloaded by a 'train-unloader', for example a so called a tippler. The unloaded material is fed as a continuous flow onto a belt conveyor line. If the material is to be stored, a proper 'stock bed' and a transport route to that stock bed have to be assigned. A stock bed is part of a stock yard and contains one material grade. The actual storage action is performed by a so called stacker.

Bulk carriers generate a transport demand from a stock bed to the ship loaders. A bulk carrier may demand batches of one or more material grades. For each batch one or more stock beds, including proper transport routes to the ship loader(s), have to be assigned. The material is retrieved from the stock bed by a machine called 'reclaimer'. In many cases stacker and reclaimer are combined in one machine, the so called stacker-reclaimer. Figure 2 shows the working of a stacker-reclaimer.

Figure 2 Stacker-Reclaimer (internet link 1)

The lay-out and operation of large scale terminals is much more complicated than shown in Figure 1. They may have several tipplers, stock yards, stacker-reclaimers, conveyors and ship loaders. Key issues in terminal operation are stock yard management, maximize utilization of equipment, dealing with equipment breakdowns, maintenance and keeping up the customer service level (Lodewijks 2003).
Train and ship (un)loading methods are discussed by Wöhlbier (1987) and Spanke (2000).

MODEL STRUCTURE AND ELEMENT CLASSES

The model is developed according the process interaction method (Zeigler 2000), providing the possibility to model very close to the real world system. The process interaction method can be summarized as follows: (1) decompose the system into relevant element classes, preferably patterned on its real world structure. An element class is characterized by its attributes. The state of each instance of a class is defined by the state or value of its attributes. (2) Determine the 'living' element classes and assign a process description to these classes making use of simulation-time consuming commands like 'hold' and 'standby' and process-interaction commands like 'interrupt', 'pause', and 'resume'. A process governs the behavior of each instance of the element class. The advantage of this modeling approach is that the model validity both structural and dynamically, can be assessed by specialists of the real system in an early stage of the project.

Next the model will be discussed and the crucial parts will be shown in pseudo code. The process-interaction approach best fits with the object-oriented paradigm.
Element classes in the model are supposed to descend from a simulation class called *SimClass*, having all the necessary properties and methods needed for process interaction modeling. Further we will use the concept of *queues* to create sets of elements. Apart from waiting line applications, the queue concept is used to create flexible object oriented data structures and to facilitate the modeling of control functions.

Job shop approach

The class of bulk terminals can be modeled very well as a modified 'job shop'. In a job shop, machine groups exist with machines having equal functions, for example a drilling group. In our model we distinguish 'Equipment line groups', containing 'equipment lines'. An equipment line consists of a sequence of pieces of equipment. It acts as a 'transport machine' that conveys a certain batch of bulk material between two points. The first machine of an equipment line always obtains the material in some way namely from a train or from a stock bed and the last machine of an equipment line delivers the material to a stock bed or a ship. The connection between the first and last machine is formed by a sequence of conveyors, a conveyor line.
Each equipment line group has its own job queue (jobQ) with 'transport jobs' with corresponding 'From' and 'To' machines.

An equipment line group is defined as:

```
TEqLineGroup     = class(SimClass)
  JobQ      :Queue
  EqLineQ   :Queue
```

Transport job

Both the arrival of a train and a ship, trigger the generation of one or more 'transport jobs'. A transport job represents a transport demand of a batch of bulk material of one grade and can have three basic origin-destination combinations:

1: from train to stock bed
2: from stock bed to ship

Eurosis-ETI. ISBN 978-77381-34-2

3: from train to ship

The train-to ship connection may use stockyard conveyors as bypass. Though the related stacker-reclaimer is blocked during the transportation period, still the stacker-reclaimer system is relieved. If the batch would have been stacked and later retrieved, a stacker-reclaimer would be busy twice as long.
Another way is using special short cut conveyors as is indicated in Figure 3 but in that case cost will be higher.
The definition of a transport job is:

```
TJob     = class(SimClass)
    Grade
    TonsToMove
    FromMachine : TEquipment // has to be assigned
    ToMachine    : TEquipment // has to be assigned
    Origin           // train or stock bed, has to be assigned
    Destination    // stock bed or ship, has to be assigned
```

Equipment and Equipment lines
In Figure 3 a terminal is shown schematically. There are 3 train unloaders (TU), 6 stacker-reclaimers (SR), 3 ship loaders (SL) and 18 conveyors (CV). We call a crossing of two conveyors, where material can be transferred from one conveyor to the other, a 'transfer point'. In practice, because of high investment costs, not every 'crossing' is a transfer point.
Three equipment line variants can be distinguished: Stacking lines, Reclaiming lines and Direct lines. Examples are:
Stacking line between TU1 and S/R1:
TU1, CV(1, 4, 13), S/R1
Reclaiming line between S/R1 and SL1:
S/R1, CV(13, 7, 10), SL1 and
Direct line between TU1 and SL1:
TU1, CV(1, 6, 18, 9, 10), SL1

Direct lines can also be formed using shortcut conveyors if available.

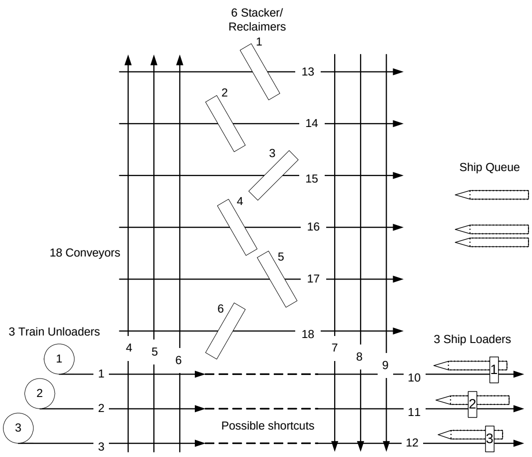

Figure 3: Schematic Terminal Lay Out with Equipment

Between the first and last machine of an equipment line in general multiple routes are possible composed of sequences of conveyors. Most equipment lines have one or more pieces of equipment in common with other equipment lines. The consequence of this is that if, for example, equipment line A is active, all other equipment lines containing one or more machines of A are not available.
The equipment line class is defined as

```
TEquipmentLine    = class(SimClass)
    EquipmentQ          :Queue        //set with all equipment needed
    EqAvailQ            :Queue        //sub set with equipment available
    MyGroup             :TEqLineGroup
    MyJobQ              :Queue        // reference to JobQ of MyGroup
    MyJob               :TJob         // job in process
    FromEq              :TEquipment  //first machine in line
    ToEq                :TEquipment  //last machine in line
    Flow                              //actual flow tons/hour
    Variant                           //Stacking, Reclaiming or Direct

  Methods
    SetEq_avail       //make own eq available for the other lines
    SetEq_Unavail   //withdraw own eq from other lines
    PauseEq           //interrupt all equipment in the eq. line
    ResumeEq        //resume process of all equipment in  the eq. line
    Process
```

The equipment class is defined as:

```
TEquipment    = class(SimClass)
    CurrentLine         :TEquipmentLine  //Eq line in which this eq is active now
    AllLineQ            :Queue           //all lines that need this equipment
    myUpDistr           :TDistribution   //distribution of Up times
    myDownDistr         :TDistribution   //distribution of down times
  Methods
    be_available        //enter EqAvailQ of all eq.lines  in AllLineQ
    be_unavailable      //leave EqAvailQ of all eq.lines exept the claiming one
    Process
```

PROCESS DESCRIPTIONS

In the processes, the next vocabulary is used:
Time consuming commands:
Hold(t) : suspend the process during t time units and proceed after that.
While (condition) standby: suspend as long as the condition is true, and then proceed.

Process-Interaction commands:
Elementx.Pause : Elementx is an element that is in a hold(t) status. Pause postpones the process of Elementx until Elementx.Resume is encountered from the process of another element. After that the remaining part of period t is continued.

'Repeat' in a process description means that the following block of indented lines is repeated continuously.
Repeat while (condition): repeat next indented block while condition is true.
Elements can be member of one or more queues at the same time.

Next the descriptions of the most relevant processes are given. Each representative of the class equipment follows a process that alternating samples a period of undisturbed work during, the 'uptime' and a down period, the 'downtime'. Every time a down period starts, the equipment piece signals that to its current equipment line. (It always has a current equipment line, because else it would not be working and consequently could not break down). Physical dynamic aspects of long heavy loaded belt conveyor systems have not been taken into account (Lodewijks 2001).

```
TEquipment.Process
 Repeat
  UpTime     =myUpDistr.Sample     //sample up time
  Hold(UpTime)                     //work during up time
  DownTime=myDownDistr.Sample //sample down time
  currentLine.PauseEq              //have current eq.line pause all its equipment
  currentLine.Pause                //pause current eq.line itself
  hold(DownTime)                   //don't work during down time
  currentLine.ResumeEq             //have current eq.line resume all its equipment
  currentLine.Resume(now)          //resume current eq.line itself at time = Now
```

An equipment line waits in its equipment line group for a job. If a job is assigned, the equipment line preserves all the machines needed in its line and works during the time needed to transport the 'tonsToMove' of its job.
Every time however when one of its machines goes down, the equipment line process is interrupted and all other machines in the line are interrupted too. If the broken machine has been repaired, all machines are resumed and the equipment line resumes to work during the remaining transport time.

```
TEquipmentLine.Process
 Repeat
  PauseEq          //interrupt all equipment of the line
  SetEq_Avail      //make eq available for all eq. lines.
  //next wait for a job AND till all eq available
  while (MyJobQ.Length=0) AND (not all eq available) Standby
  //now this line is going to work, so its equipment has to be claimed
  SetEq_Unavail //call method be_unavailable of all eq. in equipmentQ
  ResumeEq      //put all equipment to work
  MyJob:=MyJobQ.FirstElement
  MyJob.LeaveQueue(MyJobQ)
  t:=MyJob.TonsToMove/flow     //batch transport time

  if destination is a ship
    If ship is in ShipQ ship.leave(ShipQ)// ships initially wait in the ShipQ
    Hold(t)
    If ship is ready ship.resume(TNow)   // ship will leave the terminal

  if destination is a stock bed
    Hold(t)
  // in all cases update system status (tons in stock, tons transported etc)
```

Other element classes and processes

Apart from the model elements mentioned the next element classes are represented in the model

- Train generator with train arrival and load information
- Trains, consisting of a set of loaded trucks
- Ship generator with ship arrival time data and load information
- Ships, with a set of batches with grade to be loaded
- Stockyard structure with a set of stock beds and one or more associated stacker-reclaimers (of class equipment) covering the yard

The trains and ships are created during the simulation run and provided with the proper attribute values. The generation of trains and ships can be based on statistical distributions or obtained directly from historical data measured in practice on the terminal under study or another similar terminal.

Operational control

During operation, decisions have to be made at several levels and at several moments.
There are many ways to assign equipment and storage locations to a job.
In case of a train job, for example a train unloader 1) and a stock bed + associated stacker reclaimer 2) have to be assigned.

1) Train unloader assignment options: Just cyclic or according earlier planning or 'just in time' according actual data such as availability of equipment or storage space.
2) Stock bed assignment options: According planning or 'just in time' on the basis of the content of all relevant stock beds and/or the (predicted) availability and/or location of the related stacker-reclaimer and/or stacker-reclaimer work loads.

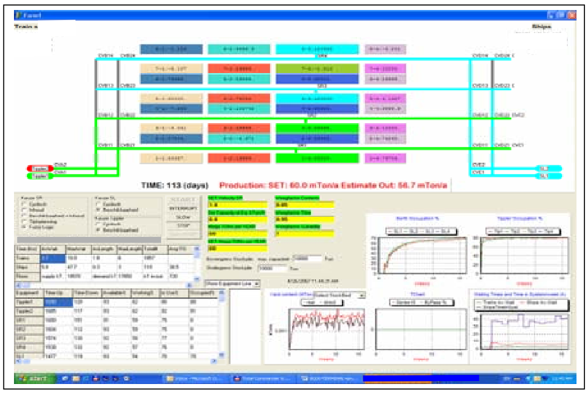

Figure 4: Screen shot of the test configuration with 4 Stock Beds per Material Grade

The same holds for the generation of transport jobs for loading a ship. A number of decision rules are built in the model and the model is made extendible for other rules.

A more strategic decision is how to arrange the stock yard lay out. Questions like: What is the influence of the bulk material throughput on the required total stock pile size and how many stock beds per grade have to be realized and of what size, have to be answered. The latter is tested in the following test case.

TEST CASE

The model is implemented in the process oriented simulation package Tomas (Veeke and Ottjes, 2000), (internet link 2). Figure 4 shows a screen shot of the model with one of the lay outs of the case. There are 8 grades involved and each grade has 4 stock bed positions available. Each stock bed is accessible by one of the 4 stacker-reclaimers. At first sight this is a logic lay-out because two trains and two ships can be served simultaneously. This is the case in Figure 4.
A disadvantage however is the fragmentation of the stacking area.

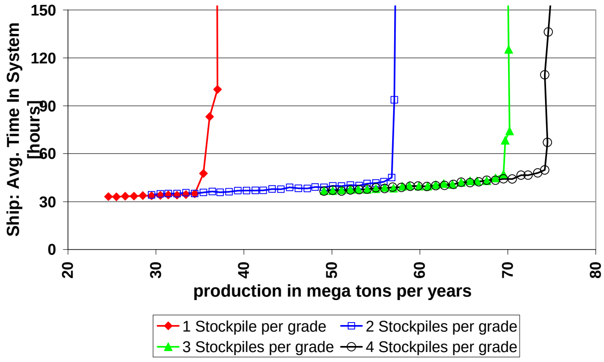

Figure 5: Results of Test Runs Varying the Number of Stock Beds per Material Grade

This lay out is compared with 3 other lay outs having 3, 2 and 1 stock beds per grade respectively. In Figure 5 the results are shown. They are as could be expected. The situation with 4 stock beds per grade performs best in terms of maximum throughput. Still it could be decided to choose the lay out with 3 stock beds per grade because of more efficient stacking area utilization.

CONCLUSIONS AND FURTHER WORK

A simulation model is discussed that can be used for design and improving operational control of large bulk terminals. The model is configurable with respect to material flow patterns, stock bed lay-out, equipment used, equipment reliability and a number of operational control methods. Break down behavior has been modeled at individual equipment level. The structure and the working of the model are explained and an example case is discussed.

Further work will focus on the development of intelligent procedures and algorithms for operational control of bulk terminals and to derive a new design approach. All results will be evaluated dynamically with simulation and validated using real world data of existing bulk terminals. The project finally has to result in a generic approach and software tools for design and control of deep sea bulk terminals.

REFERENCES

Jagerman, David; Altiok, Tayfur. (2003) "Vessel arrival process and queueing in marine ports handling bulk materials". *Queueing Syst*. 45 (2003), no. 3, 223--243.

Lodewijks, G., (2001), "Two decades dynamics of belt conveyors", *proceedings of the BeltCon 11 conference,* Johannesburg, Republic of South Africa.

Lodewijks, G., (2003). "Strategies for Automated Maintenance of Belt Conveyor Systems", *proceedings of the BeltCon 12 conference*, June 23-24, 2003, Johannesburg, South Africa.

Ottjes J. A. and Lodewijks G (2004). "Reliability of Large Scale Conveyor Systems", *Proceedings of the Industrial Simulation Conference 2004* (ISC2004) PP:324-329. June 2004. Malaga, Spain. ISBN: 90-77381-12-0

Spanke, M (2000). "Comparison of different types of ship unloaders with the focus on continuous technology". *Bulk Solids Handling*, vol. 20 Nr 4.

Veeke, Hans P.M., Jaap A. Ottjes, 2000. "Tomas: Tool for Object-oriented Modelling And Simulation". In *proceedings of Advanced Simulation Technology Conference (ASTC2000).* April 16-20, 2000, Washington, D.C. pp. 76-81. The Society for Computer Simulation International (SCS), ISBN: 1-56555-199-0

Wöhlbier, R.H. (1987), "The best of bulk solids handling, selection articles 1981-1985", *Bulk port Development & Operation.*

Zeigler B.P., Praehofer H and Kim T.G., 2000. "Theory of Modelling and Simulation 2nd Ed. Academic Press, San Diego.

Internet Link 1:
http://www.fam.cl/english/Products/Stockyard%2520systems/Stacker-reclaimers/detail.video.99999999997918.0.html

Internet Link 2: **www.tomasweb.com**