
Appendix – Simulation Design

Sjoerd van der Spoel – Chintan Amrit – Jos van Hillegersberg

University of Twente
7500 AE Enschede
The Netherlands
{s.j.vanderspoel,c.amrit,j.vanhillegersberg}@utwente.nl

There are three types of queues in the simulated terminal: a *road queue*, representing a stretch of road (e.g. between B and C); a *crane queue* (e.g. Unloading), representing a crane and the waiting line in front of that queue; and a *gate queue*, representing the truck processing (and waiting line) at the entry and exit of the terminal (e.g. gate out).

Queues The terminal is modeled as a network of queues Q , as suggested by (Alessandri, Cervellera, Cuneo, Gaggero, & Soncin, 2008; X. Chen, Zhou, & List, 2011). Each $q \in Q$ has a set of previous queues $Q_{pre} \subset Q, \forall q' \in Q_{pre} q \neq q'$. For GATE IN, Q_{pre} is the empty set $\{\emptyset\}$. Furthermore, each queue has a set of trucks q_T , which represents queued trucks. Trucks are processed in the order they arrive (FIFO).

Queue rate Each queue has a rate r_q , which determines, combined with the time needed to traverse the queue's waiting area, $\min(t_q)$ (see below), how long it will take to process a single truck. This rate is set for each individual truck, following a queue-specific distribution. The selected papers from the literature review suggest the following distributions for gate processing rate and YC processing rate:

$$r_{crane} \sim \text{Triangular}(\text{left} = 1.2, \text{mode} = 2.0, \text{right} = 3.4)$$

(Petering, 2010, 2011; Petering et al., 2009)

$$r_{gate} \sim \text{Erlang}(\mu = 2.44, k = 4) \equiv \Gamma(\theta = 0.61, k = 4)$$

(G. Chen, Govindan, Z.-Z. Yang, et al., 2013)

$$r_{road} = 0 \text{ (Only the length of the waiting area determines the queueing time)}$$

2 Enterprise Interoperability Workshop

Minimum time at queue Trucks are modeled to have a length l_T of twelve meters. Each queue also has a length $l_{q,q \in Q}$ in meters, which determines the maximum number of trucks that can be queued:

$$q_{max} = \text{floor}\left(\frac{l_T}{l_q}\right)$$

The maximum speed of trucks on our terminal is set at $V_{max}=30 \text{ km/h}=8.34 \text{ m/s}$, and trucks are assumed to have an acceleration $a_T=2 \text{ m/s}^2$ and deceleration $d_T=2.5 \text{ m/s}^2$.

Combined with the queue length l_q , this gives the minimum time to traverse the length of a queue. Consider a truck to start the queue at t_0 with speed zero, accelerates until it reaches the maximum speed, and then decelerates to arrive at the end of the queue at t_{end} with speed zero. The point in time where it reaches V_{max} is labeled B, the point where it has to start decelerating to stop in time is labeled C. This is shown in Figure 1.

The time between t_{start} and t_{end} is the minimum queuing time $\min(t_q)$. It is the minimum, as this only occurs when there is just one truck in the queue, otherwise, it would have to wait for other trucks to clear the queue. The minimum queuing time is calculated as follows:

$$B = \frac{V_{max}}{a} \quad (1)$$

$$s = \text{Distance covered} \quad (2)$$

$$s(t_0 \rightarrow B) = \frac{1}{2} * a_T * B^2 \quad (3)$$

$$s(C \rightarrow t_{end}) = \frac{1}{2} * d_T * \frac{V_{max}^2}{d_T} \quad (4)$$

$$\text{if } s(t_0 \rightarrow B) + s(C \rightarrow t_{end}) < l_q : \quad (5)$$

$$s(B \rightarrow C) = l_q - (s(t_0 \rightarrow B) + s(C \rightarrow t_{end})) \quad (6)$$

$$C = B + \frac{s(B \rightarrow C)}{V_{max}} \quad (7)$$

$$t_{end} = \min(t_q) = C + \frac{V_{max}}{d_T} \quad (8)$$

$$\text{if } s(t_0 \rightarrow B) + s(C \rightarrow t_{end}) \geq l_q : \quad (9)$$

$$h = \sqrt{\frac{2 * l_q}{\frac{1}{a_T} + \frac{1}{d_T}}} \quad (10)$$

$$P = \frac{h}{a_T} \quad (11)$$

$$e = h + d_T * P \quad (12)$$

$$t_{end} = \min(t_q) = \frac{e}{d_T} \quad (13)$$

Arrival at gate The selected papers from the literature review suggest four different distributions for the amount of trucks arriving at GATE IN: either a $\beta(\alpha = 1.25, \beta = 3.25)$ distribution (G. Chen, Govindan, & Z. Yang, 2013), an exponential distribution (G. Chen, Govindan, Z.-Z. Yang, et al., 2013), a uniform distribution (G. Chen, Govindan, & Z. Yang, 2013; Namboothiri & Elera, 2008), or a Poisson distribution (Zehendner & Feillet, 2014).

The Poisson and B distribution are both used in a context where either trucks arrive to pick up containers within a certain interval/window after a ship has arrived at the terminal, or the trucks have an appointment (G. Chen, Govindan, & Z. Yang, 2013; Zehendner & Feillet, 2014). Chen *et al.* and Namboothiri and Elera suggest that the arrival of trucks

4 Enterprise Interoperability Workshop

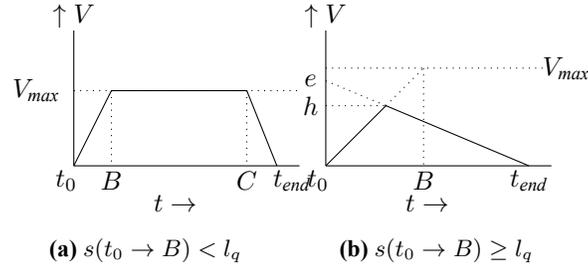
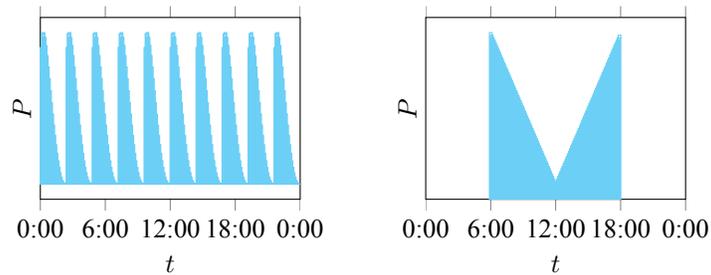


Figure 1. Speed of a truck on a queue over time. B is the point in time at which the truck reaches its maximum speed. C is the point in time when the truck has to start decelerating to be at a standstill at the end of the queue. Given a truck's acceleration and deceleration, and the length of the queue (distance), the total time to cover the queue can be calculated as shown by the formulas on page 3.

throughout the day has multiple peaks, such as early morning and early afternoon (G. Chen, Govindan, Z.-Z. Yang, et al., 2013; Namboothiri & Erera, 2008). So, literature suggests two variants for truck arrivals with four distributions: ship-arrival based 'windows' with a β or Poisson distribution and 'peaks' with a uniform or exponential distribution.

Chen *et al.* are most specific about truck distribution, as they also give the parameters of the β distribution (G. Chen, Govindan, & Z. Yang, 2013). Therefore, we have implemented the 'window' variant with a β distribution. For the 'peak' variant, Namboothiri and Erera (2008) provide specifics for truck arrival distribution (Namboothiri & Erera, 2008), whereas Chen *et al.* (G. Chen, Govindan, Z.-Z. Yang, et al., 2013) only mention that truck arrival is distributed exponentially. Namboothiri and Erera(2008) describe three scenario's: uniform, where the same amount of trucks arrive throughout the day; midday-heavy, where three times more trucks arrive around the middle of the day than at the start and end; and morning-afternoon-heavy, where three times more trucks arrive at the start and end of the day than at the middle of the day. Note that their research is about capacity requested by a trucking company, rather than actual arrivals, but we assume that these would follow a similar distribution. Figures 2a and 2b show both implemented distributions.

Pseudocode Algorithm 1 shows pseudocode for the main element of our simulation, which was implemented in Python. This algorithm is run once every second of the duration. Line 4 refers to a function that generates a truck with a random set of 'orders': it can unload an empty container and then load a full one, or only either unload or load. In line 5, the queue checks if the current truck has been in the queue longer than the queue rate q_r . Lines 9 and 11 show the mechanism for dealing with full queues. If the the next queue



(a) $B(1.29, 3.25)$ probability density function for truck arrivals, with 10 'windows' of 144 minutes
(b) Morning-afternoon-heavy probability density function for truck arrivals, with three times more trucks at the start and end of the day than at midday

Figure 2. Probability density functions for truck arrivals at the terminal

(a road, crane, or gate) for a truck is completely full, it has to stay at the queue where it is and that queue stops processing. As soon as a queue manages to send a truck to a next queue (or the end of the terminal), the queue notifies all previous queues that they can start processing again. Every time a truck leaves the terminal, its start time, end time, and orders are recorded in a dataset.

6 Enterprise Interoperability Workshop

```
1  $t \leftarrow 0$ ;  
2 for  $t \leftarrow 0$  to simulation duration do  
3    $N_t \sim$  truck arrival distribution;  
4   Generate  $N_t$  trucks;  
5   foreach queue in queues do  
6     if queue is running and  $|\text{queue}| \neq 0$  and  $t_{\text{queue}[0]} > r_{\text{queue}}$  then  
7       if next queue is not full then  
8         Move first truck in queue to next queue;  
9         Notify  $Q_{pre}$ ;  
10      else  
11        Stop queue;  
12      end  
13    end  
14  end  
15 end
```

Algorithm 1: Pseudocode for the simulation