14 Warehouse Math

René de Koster, Rotterdam School of Management, Erasmus University, rkoster@rsm.nl

Abstract

After Jo's unexpected death in Vancouver, I decided to devote this chapter to four of his favorite topics: math, models, information systems, and warehouses. Jo also very much liked popularizing the logistics area. In an attempt to combine all this in one chapter I show that elementary math models reveal powerful insights that can help understand better design and management decisions in facility logistics operations. Subsequently I develop models for designing a truck-operated unit-load warehouse (sizing it, calculating the load orientation, and determining the warehouse length-width ratio). Next the optimal stacking depth is calculated and for automated unit-load warehouses the optimum rack size, number of cranes and optimum storage zone boundaries are determined. Finally the impact of the number of inducts on sorting system throughput is determined.

14.1 Introduction

Warehouses and distribution centres play an important role in the supply chains of companies as they decouple supply from demand. Moreover, warehouses have a role in forming an assortment and as the prime point in the supply chain to group material flows to create transport efficiencies. Companies have moved their push-pull boundaries as far downstream as possible, implying many warehouses are involved in value-adding activities. According to the report of HIDC and TNO (2009), warehousing activities (including value addition) directly contributes €15.7 billion annually to the Dutch GDP. Indirect services (like material handling systems, consultancy, leasing services) are not included in that number.

In an attempt to minimize operational cost many companies operating warehouses have invested in largely automated warehouse operations. Storage of unit loads was already automated in the sixties with automated storage and retrieval systems (AS/RS), usually employing high-bay racks with rail-mounted cranes. Automation of the order picking process of individual selling units is known since about 15 years in sectors with limited variety in product shapes, such as pharmaceuticals. In the last decade it became possible to automatically store, replenish and order pick dry groceries from a fairly large and complicated assortment. The warehouses of Edeka and Lidl in Germany are the first in the grocery retail sector to receive, store, order pick, sequence products, load roll containers and prepare loads for shipments in largely robotized warehouses. Scarcity of labour, 24-hour operations, an urge to work in relatively smaller (and hence cheaper) facilities, sufficiently large shipping volumes per facility, and cheaper technologies have driven these developments.

In order to design, plan, and operate such facilities, models are needed. Software algorithms taking care of order grouping and release, storage slot allocation, pick location allocation, timing of replenishments between reserve and forward storage locations, balancing work loads over pick stations, planning shipments, sequencing pick instructions, and routing order pickers are used to realize the desired performance. These algorithms are the fruits of academic research. Although models for certain configurations can easily become large and intractable, it is possible to derive simple models that yield much insight in system performance. As examples I will treat design models focusing on *unit load* warehouses. Unit loads are pallets, totes, roll cages, or any other product carrier handled as a single unit. Unit-load storage systems can be found in many warehouses. They are used to store inbound materials, which are later moved to forward areas or pick stations where the orders for single cases or units are picked. Figure 1 sketches product flows in a typical warehouse, where the "reserve storage" area represents the unit-load warehouse.

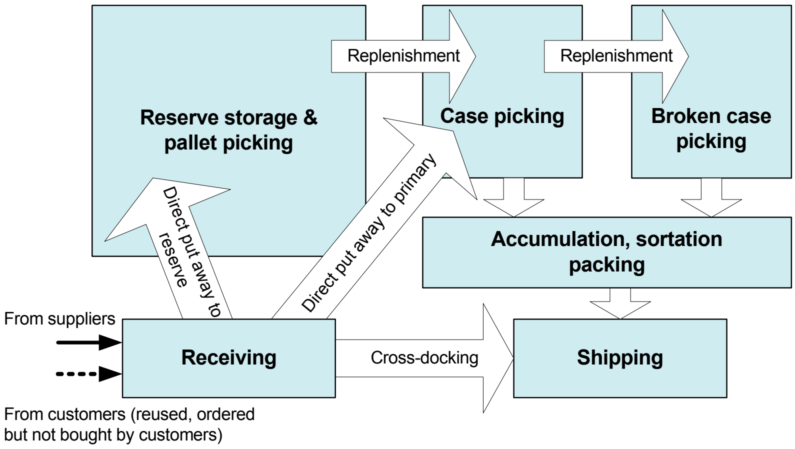

Figure 1. Typical warehouse functions and flows (Tompkins et al. 2003)

In the next sections the following warehouse-design questions are covered subsequently, using elementary models.

Rack-based unit-load warehouse design:
- How many unit loads should the warehouse cater for?
- Should unit loads be stored deep or wide?
- What is the optimum size (length/width ratio) of a unit-load warehouse?

Block-stack based unit load warehouse design:
- What is the optimum storage depth in a multi-deep block stack?

AS/RS-based unit-load warehouse design:
- What is the optimum size (length/height ratio) of a unit-load storage rack in an AS/RS?
- How many storage/retrieval cranes are needed to achieve a certain throughput?
- What are the optimum zone boundary shapes for turnover-class based storage?

Sorting system design:
- How many inducts should the sorter have?

14.2 Rack-based unit load warehouse design

The first question asks how many loads should be stored in the warehouse. Only then can the warehouse and the racks be sized. We treat these two topics in the following subsections.

14.2.1 Determining the number of unit loads to be stored

The number of loads the storage system should be able to cater for depends on the following factors:

- The number of products, N, stored (average and maximum simultaneously). Every product, even when stored in a small quantity needs a separate storage slot, capable of storing the unit load. In determining N not only current products should be included, but also the assortment that should be catered for in the medium-term future.
- The number of slots needed per product.
- The size and weight of the products: how many product units fit on the unit load?
- The amount of slack the storage space should have. Every warehouse needs slack space to warrant a smooth operation. According to a rule of thumb explained by Tompkins et al. (2003, p.403), the slack space β should be around 20%. The slack space depends on the technology used. In more automated warehouses less slack space is needed.
- The used storage strategy. In case a *dedicated* storage strategy is followed every product, or even stronger, every unit load of a product, has a fixed storage location, which usually is based on turnover speed of the load. In case of a *shared* storage policy unit loads can share storage slots, albeit not at the same point in time. In practice combinations are often used, for example a fixed slot for the first load and shared slots for the remaining loads of all products.

The number of loads to be stored per product depends on its replenishment policy. Assuming a continuous review <*s* ,*Q*> ordering policy, with s the reorder level and Q the economic order quantity, the reorder level s will have to cover the mean demand over lead time plus a safety factor, depending on the desired service level of the product, times the standard deviation of the demand over lead time. In formula, $s = \mu_d \cdot \bar{L} + Z_\alpha \sigma_{dL}$, where μ_d is the product's mean demand per day, L is the (stochastic) supply lead time in days, Z_α is the safety factor for a service level (fill rate) 1-α of the product. The term $Z_\alpha \sigma_{dL}$ is called the safety stock, *ss*. In case lead time is constant and demand is iid, then the standard deviation of demand over lead time, σ_{dL}, equals $\sigma_d \sqrt{L}$, with σ_d the standard deviation of daily demand. The safety factor can be determined if the product's demand distribution is known. The economic order quantity Q equals $\sqrt{2DK}$, where D is the annual demand and K is the ratio of ordering over annual unit inventory holding cost. In an <*s, Q*> system, it is straightforward to see the average stock of a product equals $\frac{Q}{2} + ss$ and the maximum stock equals $Q + ss$. The total number of storage slots C needed will therefore fall between

$$\sum_{i=1}^{N} \left\lceil \frac{Q_i}{2} + ss_i \right\rceil (1+\beta) \text{ and } \sum_{i=1}^{N} \left\lceil Q_i + ss_i \right\rceil (1+\beta), \tag{1}$$

where all quantities are expressed in unit loads and β is the system's slack factor. The notation $\lceil x \rceil$ indicates x should be rounded upwards (to full unit loads). Although this analysis partly answers the first question, some intricate further questions remain. The gap between the lower and upper bound may be substantial: think of a warehouse for 1,000 products where all products are replenished in lots of 2 pallets, and the average space needed is 1 pallet per product. Should the warehouse accommodate for 1,200 or 2,400 pallets? As in most practical environments some degree of storage slot sharing is possible, the number of slots needed is usually closer to the lower than to the upper bound.

14.2.2 Determining the unit load storage orientation

The footprint of most unit loads is not square, but rectangular, meaning they have a long and short side. Euro pallets and block pallets have a size of 120×80 and 120×100cm, respectively. Euro-sized totes have a footprint of 60×40cm. As most loads can be retrieved from any direction, an important design decision is whether loads have to be stored wide or deep.

In conventional unit load warehouses the building, with fixed installations, may take up to 70% of the total operational costs (rent/interests, write-offs, maintenance, energy; De Koster 1996). The most important criterion for load orientation therefore is space consumption. Figure 2 sketches the space occupied by two loads stored on opposite sides of a travel aisle, stored wide or deep. In the calculation we assume the loads are Euro pallets, and the truck we use is a pallet stacker (a small type of reach truck, capable of lifting about 4*m* high) which needs a turning circle of 2.1*m*.

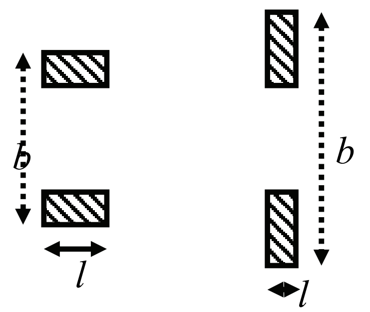

Figure 2. Two unit loads stored at opposite sides of the travel aisle, stored wide (left) or deep (right)

In case the loads are stored wide, the aisle width+two pallet depths+safety space equals b= 2.1+ $2\times(0.8+0.2)$ =4.1m. Therefore, the space occupied by two pallets plus aisle, O_w becomes $b\times l$= $4.1\times(1.2+0.2)$ = 5.7m^2. In case loads are stored deep, the net aisle width becomes 2.5m, since the pallet has now become 0.4m longer, leading to a space of O_d =$5.3\times(0.8+0.2)$= 5.3m^2, implying a space saving of 7%. This explains why loads in unit load facilities should always be stored deep; if not, this calls for a serious inquiry.

14.2.3 Determining the length and width of the warehouse

Figure 3 shows two different layouts of the warehouse, the left one has a single input/output (I/O) point whereas the right one has split input and output points. We aim to determine the length L and the width B of the warehouse, under the assumption the storage capacity C is given (from Equation (1)). We assume the storage strategy employed in the warehouse is random (i.e. each location is equally likely to contain the requested unit load), storage and retrieval requests arrive on line and the warehouse truck used to store and retrieve unit loads carries out single cycles only. In case of a storage job, such a single cycle consists of fetching the load at position X, storing it at some storage location Y and returning empty to location X. The cycle is similar in case of a retrieval. We furthermore assume for now the number of storage levels n is given, hence we divide C by n to obtain the capacity per level. In optimizing L and B, we have to realize warehouse trucks do not drive and lift simultaneously for safety reasons, i.e. they always -should- drive with the forks in the lowest position. Upon arrival at the storage location, the truck makes a 90º curve and starts lifting until it reaches the desired slot and stores or retrieves the load. After this it lowers its forks and travels to the output point. Therefore, the only element of the storage/retrieval cycle influenced by B and L is the travel time. In both layouts, the mean travel time to carry out a single cycle equals (approximately) $B+\frac{L}{2}$.

This is to be minimized under the condition $L \cdot B \geq C$. Substituting the constraint into the objective, and applying first order conditions yields L=2B. In other words the warehouse should be twice as wide as deep. If the number of unit loads to be stored P is given, it is now straightforward to calculate the real warehouse dimensions, since $P \approx 2n\frac{L}{l}\frac{B}{b}$. Note that $\frac{B}{b}$ represents the number of aisles in the warehouse and $\frac{L}{l}$ represents the number of storage sections per aisle side. Combining this with L=2B yields: $B=\frac{1}{2}\sqrt{\frac{Plb}{n}}, L=\sqrt{\frac{Plb}{n}}$. (2)

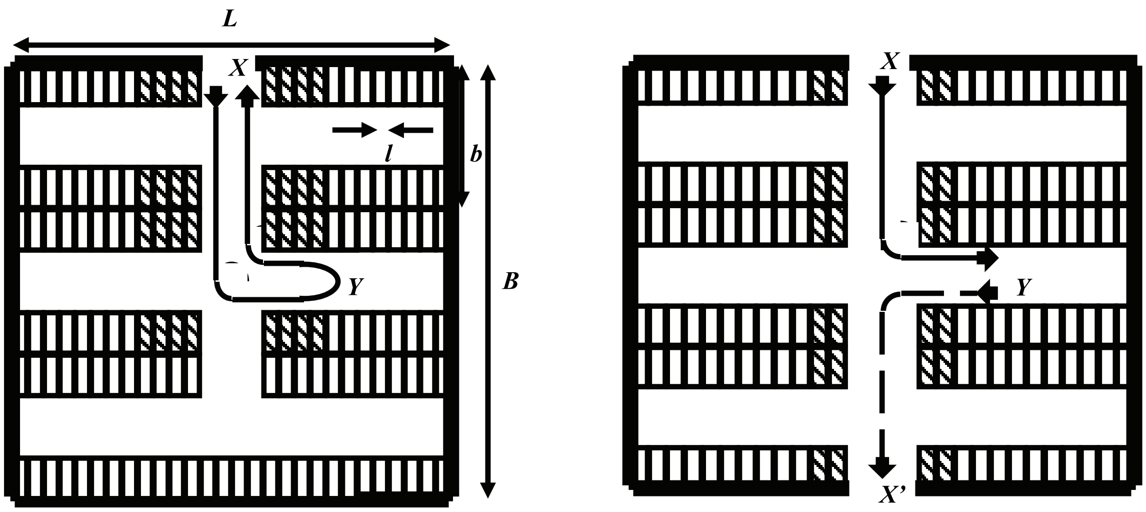

Figure 3. Top view of two different layouts of the unit load warehouse

In determining the real L and B values from (2) we have to round the values found such that $\frac{B}{b}$ and $\frac{L}{l}$ are integers. In practice this is not a great restriction as the cycle travel time is greatly insensitive to the exact warehouse shape ratio, as shown in Figure 4. This may explain why so many warehouses in practice have suboptimal shape factors. I suspect though there may be other explanations.

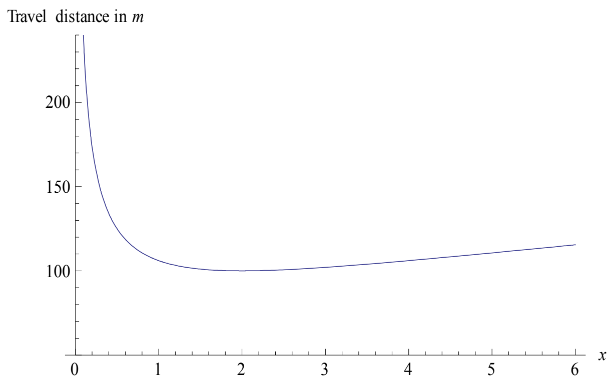

Figure 4. Travel distance $\sqrt{\frac{C}{x}}+\frac{1}{2}\sqrt{Cx}$ as function of the warehouse length/width ratio x, for a warehouse of C=5000m^2.

The remaining question is how to determine the number of storage levels n. The number of levels is primarily determined by the building costs and the number of trucks and people needed. In general driving a truck is much speedier than lowering or lifting, implying cycle times go up for higher buildings. Therefore, for a given throughput we might need more trucks and people in a high building. However, high buildings are much cheaper than low buildings of the same cube. In view of the dominant part of building-related costs within the operational costs higher buildings are usually to be preferred. However, in order to find the exact trade-off curve and determine the optimal value of n, the precise real costs should be calculated as function of n. An impression on how to do this can be found in De Koster (1996).

14.3 Block-stack based unit load warehouse design

Many warehouses store bulk unit loads in block stacks, drive-in or drive-through racks. Such block stacks are particularly attractive if few products need to be stored in large quantities per product. Think of beer and soft-drink manufacturers or retail warehouses storing such products. Block stacks are usually arranged around transport aisles and have a fixed depth (the lane length). A major question in such stacks is to determine the proper storage lane length. If lanes are too deep, the average space usage will be very poor (this effect is called 'honeycombing', see Tompkins et al. 2003). However, if lanes are short, we may need multiple lanes per product which increases warehouse size. The main objective is again to minimize the space usage to store all unit loads needed. Figure 5 sketches a block stack with 3-deep lanes accessible from two sides (like using drive-through racks).

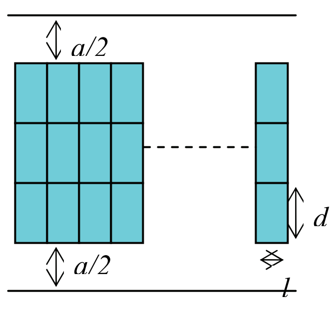

Figure 5. Top view of a 3-deep block stack lane, with access on two sides, through travel aisles of width *a*. The load has size $d \times l$ (including safety space)

Our prime objective is to determine the number of unit loads, x, to be stacked behind each other in a lane. We assume product i is received in the warehouse in batches of q_i unit loads.

For a given lane of depth x (in number of unit loads) the average total space occupied by all products plus half of the two aisles equals $\left(dx+a\right)\cdot l\sum_{i=1}^{N}\left\lceil \frac{q_i}{2x}\right\rceil$, which approximately equals $\left(dx+a\right)\cdot l\sum_{i=1}^{N}(\frac{q_i}{2x}+\frac{1}{2})$. Applying first order conditions yields $x=\sqrt{\frac{a}{dN}\sum_{i=1}^{N}q_i}$. If products can be stacked on top of each other (say n_i levels for product i), we have to divide q_i by n_i in this formula. If access is possible from one side only (like in the case of drive-in racks) we have to divide a by 2.

14.4 AS/RS-based unit-load warehouse design

In an AS/RS (like a miniload, or high-bay pallet warehouse) each storage aisle has its own aisle-bound crane. We first calculate the cycle time (time to store or retrieve a load) of the crane assuming a random storage strategy and use this to determine the optimum size (length/height ratio) of a unit-load storage rack in an AS/RS. From this we can determine the number of cranes (and aisles) and the we can find the optimum zone boundary shapes.

14.4.1 Calculating the optimum rack shape

In order to calculate the optimum rack shape we have to realize a crane drives and lifts simultaneously. The crane speeds are v_x and v_y, in horizontal and vertical direction, respectively.

The driving time to the farthest location is now $t_x=\frac{L}{v_x}$; the lifting time to the highest location $t_y=\frac{H}{v_y}$ (see Figure 6). We aim to find $E[W]$, where W is the driving time to a random location (X,Y), and back to the depot.

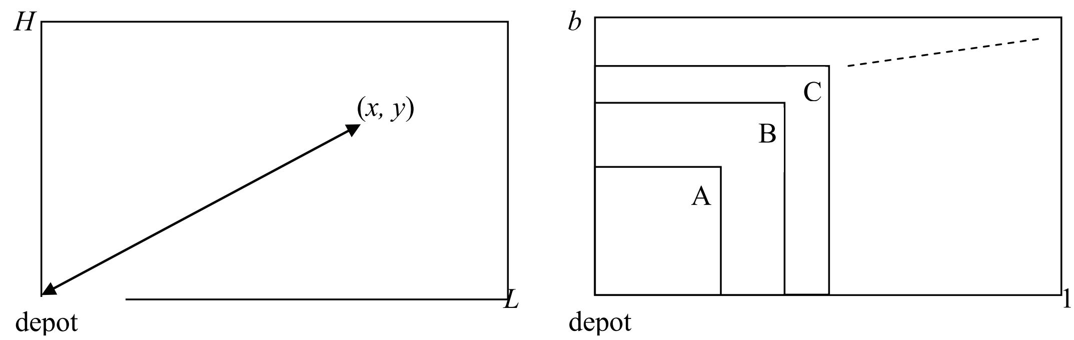

Figure 6. Side view of the rack, with the depot in the lower left corner. Right part: normalized rack

In order to simplify the calculations, following Tompkins et al. (2003), we define $T=\max\{t_x,t_y\}$, $b=\min\{\frac{t_x}{T},\frac{t_y}{T}\}$. We assume the rack is longer (in time) than it is high (i.e. $t_x \geq t_y$ and we normalize the rack by dividing the time dimensions by T. In the normalized rack (see Figure 6), the horizontal storage location X and vertical storage location Y are now stochastic variables uniformly distributed on [0,1] and [0,b], respectively. Let $Z=\max\{X,Y\}$, then $E[W]=2E[Z]\cdot T$. Since the crane drives and lifts simultaneously, $F_Z(z)\equiv P[Z\leq z]=P[\max\{X,Y\}\leq z]=P[X\leq z]\cdot P[Y\leq z]$, with

$P[X \le z] = \begin{cases} z, \text{if } 0 \le z \le 1 \\ 1, \text{if } z > 1 \end{cases}$, $P[Y \le z] = \begin{cases} \frac{z}{b}, \text{if } 0 \le z \le b \\ 1, \text{if } z > b \end{cases}$. In other words

$$F_Z(z) = \begin{cases} z^2 / b, & \text{if } 0 \le z \le b \\ z, & \text{if } b < z \le 1 \\ 1, & \text{if } z > 1 \end{cases}.$$

By taking the derivate of F_z we find the probability density function f_z and from this we can obtain the expected value of Z by calculating $E[Z] = \int_0^\infty z f_Z(z) dz$. From $E[Z]$ we find

$$E[W] = (1 + \frac{b^2}{3})T \tag{3}$$

As an example, assume the rack is 50 sec long and 50 sec tall, then b=1 and T=50. The driving time to a random location and back to the depot is therefore 66.7 sec (and not 50 sec!). We can now use formula (3) to optimize the rack shape as follows: $\min! E[Z] = (1 + \frac{(t_y / t_x)^2}{3}).t_x$, s.t. $t_x \cdot t_y = C$. Substituting the constraint into the objective and applying first order conditions yields $t_x = t_y (= \sqrt{C})$. In other words, b=1; the optimum rack is square *in time*. However, since cranes usually drive much faster than they lift (in a ratio of about 4 to 1), optimum racks should be rectangular with this same ratio (of 4 to 1). It is fairly easy to see from the outside of a high-bay warehouse whether it (approximately) has this optimal shape. The building should be about 4 times longer than the height.

14.4.2 Calculating the number of cranes

Cranes are very expensive. Therefore we use formula (3) to calculate the minimum number of cranes needed, for a given storage capacity (using formula (1)) and throughput, expressed in number of unit loads to be stored and retrieved per time unit. As an example assume we need 4000sec^2 of storage space and a throughput of 200 pallets per hour. The problem can be formulated as: $\min! N$, such that $2N \cdot t_x \cdot t_y = 4000$, and $\frac{N \cdot 3600}{(1 + \frac{(t_y / t_x)^2}{3})t_x + 2\delta} \ge 200$. δ is the -constant- load pick-up or drop-off time, about 18 sec in pallet warehouses. This problem can easily be solved iteratively by starting with 1 crane, calculating the resulting (square) rack size and then calculating the throughput using formula (3). If the throughput constraint is met we are done, otherwise the number of cranes (and aisles) is increased by 1, etcetera. For the example given, the optimum number of cranes and aisles is 4, leading (for δ=18) to a throughput of 218.8 pallets/hour.

14.4.3 The optimum storage-zone shape

Many warehouses do not apply random, but class-based storage. The idea is to divide the products in turnover-based classes, with the fast-moving products (A-products) stored close to the depot and slower-moving products farther away. Within a storage class, products are stored randomly. The question is what the optimum storage zone looks like. It is fairly easy to see the optimum shape of a storage zone is a square L-shape in the time dimension, as sketched in

Figure 6 (right part). Since the crane drives and lifts simultaneously, all points on the boundary of such an L-shape have exactly the same travel time. Therefore they must belong to the same travel time class. In conclusion, the optimum zone shapes are rectangular in meters, in a ratio of about 4 to 1. Although this might seem fairly obvious I had a recent visit with students to a high-tech fully automated carpet storage warehouse. The carpet rolls were stored vertically, retrieved by an overhead crane moving in *x*- and *y*-direction simultaneously. The zone boundaries were not square in time, but circular around the depot.

14.5 The impact of the number of sorter inducts on throughput

Many warehouses pick orders in batch and sort them per customer order using sorting machines. Figure 1 gives an impression of the position of this process in the total product flow. Sorting machines have strongly improved over the last decades; they have become faster and cheaper, more ergonomic, less noisy, and more generally useable. Figure 7 shows pictures of a loop sorter, where products can circulate if the destination chute is full.

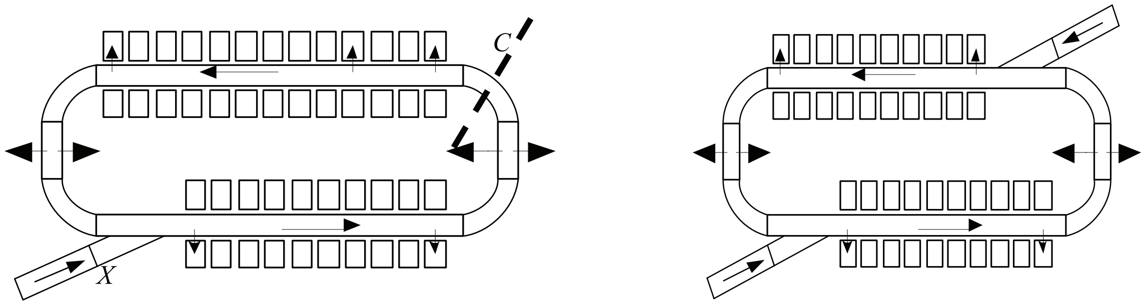

Figure 7. Loop sorter sorting to two sides with one (left) and two inducts (right). Source: Vanderlande

The sorter has line capacity *C* (products per time unit) at any intersection. However, the number of products actively sorted by the sorter can be larger than *C*, depending on the number of inducts the sorter has. Assume the sorter has *N* inducts, equally spaced around the sorter, with the sorting chutes equally divided along the sorter. The maximum inbound flow at a sorter induct *X*, is now limited by the products on the sorter fed by other inducts, but not yet sorted. For two inducts with flows *X* and *Y* we find:

$X+\frac{1}{2}Y=C$ and $Y+\frac{1}{2}X=C$, implying *X*=*Y*. Therefore $X=\frac{2}{3}C$. The total throughput capacity of the sorter is therefore $2X=4C/3$, or 33% more than the line capacity, achieved with one induct. In general, with *N* inducts we find $\frac{1}{N}X\sum_{i=1}^{N}i=C$, or $X=\frac{2}{N+1}C$. Hence the throughput is $NX=\frac{2N}{N+1}C$. Note that for large *N*, we can nearly double the sorter's line capacity! This also works in practice. During a company tour at a discount retailer, it was mentioned the sorter was a bottleneck for further store expansion. The sorter had three inducts clustered together, effectively acting as one induct. After making a remark on this, I noticed the next year the sorter had not been upgraded, but a new induct had been bought, placed diagonally opposite the first induct group. This had substantially increased sorting capacity.

14.6 Conclusions

The above examples show elementary math models can be used to provide insight into warehouse design questions. Many extensions of the above-mentioned models exist, albeit often with more complicated analyses.

References

De Koster, M.B.M. (1996). Ontwerp van een eenvoudig magazijn. In: J.P. Duijker et al. (Eds.), Praktijkboek Magazijnen/Distributiecentra (pp. 3.10A.01-3.10A.20). Deventer: Kluwer.

HIDC, TNO (2009). De Logistieke Kracht van Nederland 2009. Zoetermeer: HIDC

Tompkins, J.A., White, J.A., Bozer, Y.A., Tanchoco, J.M.A. (2003). Facilities Planning. New York: Wiley.