Georgia Tech College of Engineering
H. Milton Stewart School of
Industrial and Systems Engineering

ISyE 6339 Physical Internet Engineering

Casework 1.1

BotWorld's Dedicated European Supply Chain Design

Guillaume Eymery - Louise Lacroix - Tiankuo Zhang

H. Milton Stewart School of Industrial and Systems Engineering

Contents

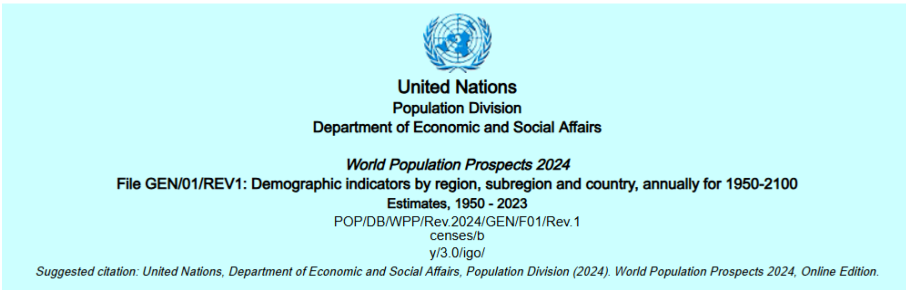
List of Figures

List of Tables

1 TASK 1: Market Demand Forecast & Scenario Generation

1.1 Estimation of the 2026 - 2033 population of targeted countries and capitals, metropolitan areas

In order to complete this task, we used the data provided by the Department of Economic and Social Affairs of the United Nations Population Division. Filtering the European



United Nations
Population Division
Department of Economic and Social Affairs

World Population Prospects 2024
File GEN/01/REV1: Demographic indicators by region, subregion and country, annually for 1950-2100
Estimates, 1950 - 2023
POP/DB/WPP/Rev.2024/GEN/F01/Rev.1
censes/b
y/3.0/igo/
Suggested citation: United Nations, Department of Economic and Social Affairs, Population Division (2024). World Population Prospects 2024, Online Edition.

Figure 1: World Population by UN

Countries targeted by BotWorld, we obtain the following table:

Region	Year	Projected Population (thousands)
France	2026	66,701
France	2027	66,792
France	2028	66,884
France	2029	66,973
France	2030	67,064
France	2031	67,152
France	2032	67,238
France	2033	67,326
Germany	2026	83,741
Germany	2027	83,547
Germany	2028	83,338
Germany	2029	83,120
Germany	2030	82,891
Germany	2031	82,671
Germany	2032	82,434
Germany	2033	82,214

Table 1: Extract of Projected Population per Countries (2026-2033).

By cross-referencing the projection table with the percentage of the population of each country living in the selected agglomeration areas:

Name of Capital/Metropolitan Area	Country	Percentage of Population (%)
Paris	France	18.6
Lyon	France	3.6
Marseille-Aix-en-Provence	France	2.8
Toulouse	France	2.1
Lille	France	1.8
Nice-Côte d'Azur	France	1.5
Nantes	France	1.4
Bordeaux	France	1.8
Strasbourg	France	1.3
Berlin	Germany	8.3
Hamburg	Germany	2.2
Munich	Germany	2.3
Cologne	Germany	1.9
Frankfurt	Germany	1.3
Stuttgart	Germany	1.2

Table 2: Percentage of Population in Capital and Metropolitan Areas (extract).

We obtain the final table of population projections.

Metropolitan Area	2026	2027	2028	2029	2030	2031	2032	2033
Paris	12,406,386	12,423,312	12,440,424	12,456,978	12,473,904	12,490,272	12,506,268	12,522,636
Lyon	2,401,236	2,404,512	2,407,824	2,411,028	2,414,304	2,417,472	2,420,568	2,423,736
Marseille-Aix-en-Provence	1,867,628	1,870,176	1,872,752	1,875,244	1,877,792	1,880,256	1,882,664	1,885,128
Toulouse	1,400,721	1,402,632	1,404,564	1,406,433	1,408,344	1,410,192	1,411,998	1,413,846
Lille	1,200,618	1,202,256	1,203,912	1,205,514	1,207,152	1,208,736	1,210,284	1,211,868
Nice-Côte d'Azur	1,000,515	1,001,880	1,003,260	1,004,595	1,005,960	1,007,280	1,008,570	1,009,890
Nantes	933,814	935,088	936,376	937,622	938,896	940,128	941,332	942,564
Bordeaux	1,200,618	1,202,256	1,203,912	1,205,514	1,207,152	1,208,736	1,210,284	1,211,868
Strasbourg	867,113	868,296	869,492	870,649	871,832	872,976	874,094	875,238
Berlin	6,950,503	6,934,401	6,917,054	6,898,960	6,879,953	6,861,693	6,842,022	6,823,762
Hamburg	1,842,302	1,838,034	1,833,436	1,828,640	1,823,602	1,818,762	1,813,548	1,808,708
Munich	1,926,043	1,921,581	1,916,774	1,911,760	1,906,493	1,901,433	1,895,982	1,890,922
Cologne	1,591,079	1,587,393	1,583,422	1,579,280	1,574,929	1,570,749	1,566,246	1,562,066
Frankfurt	1,088,633	1,086,111	1,083,394	1,080,560	1,077,583	1,074,723	1,071,642	1,068,782
Stuttgart	1,004,892	1,002,564	1,000,056	997,440	994,692	992,052	989,208	986,568

Table 3: Population Data for Major Metropolitan Areas (2026-2033).

1.2 Forecast of the expected annual demand for the entire product line and for each product

Using the data computed in Task 1.1 and cross referencing it with WorldBot's market penetration strategy and projected sales:

- 2026: France, Germany
- 2027: Austria, Belgium, Ireland, Italy, Luxembourg, Netherlands, Portugal, Spain, Switzerland, United Kingdom
- 2028: Denmark, Estonia, Finland, Latvia, Lithuania, Norway, Poland, Sweden
- 2029: Bulgaria, Croatia, Republic of Cyprus, Czech Republic, Greece, Hungary, Malta, Romania, Slovakia, Slovenia

Year	Optimistic (99%)	Most Probable	Pessimistic (99%)
1	0.05%	0.025%	0.02%
2	+0.025%	+0.02%	+0.01%
3	+0.02%	+0.01%	+0.005%
4+ (yearly)	+0.01%	+0.005%	-0.002%

Table 4: Growth projections with optimistic, most probable, and pessimistic scenarios.

We obtain the projected sales for WorldBot per Metropolitan Area and per year.

We can observe that metropolitan areas such as Graz or Brussels have a projected demand for 2026 of 0. This makes sense considering that WorldBot has not penetrated the Austrian and Belgian markets yet.

		2026		
Area		Optimistic (99%	Most Probable	Pessimistic (99%
Paris	France	6203	3101	2481
Lyon	France	1200	600	480
Marseille-Aix-en-Provence	France	933	466	373
Toulouse	France	700	350	280
Lille	France	600	300	240
Nice-Cote d'Azur	France	500	250	200
Nantes	France	466	233	186
Bordeaux	France	600	300	240
Strasbourg	France	433	216	173
Berlin	Germany	3475	1737	1390
Hamburg	Germany	921	460	368
Munich	Germany	963	481	385
Cologne	Germany	795	397	318
Frankfurt	Germany	544	272	217
Stuttgart	Germany	502	251	200
Vienna	Austria	0	0	0
Graz	Austria	0	0	0
Linz	Austria	0	0	0
Brussels	Belgium	0	0	0
Antwerp	Belgium	0	0	0
Ghent	Belgium	0	0	0

Figure 2: Demand Projection for 2026 (extract)

1.3 Expansion of the annual forecasts into more granular weekly and daily global and model-specific demand over the 2026-2033 horizon

For the next tasks, we will use Python in to iterate faster and have more control over the data.
The first objective is to import the necessary information: the sheet from Task 1.2 (`.csv` file) and the product data with the demand share and price per model (which is stored in a pandas DataFrame).
To do so, we implemented a function called `calculate_demand()` that take into parameter the previous data and outputs a pivot table (in the format of a DataFrame) showing for one given year in each city, the Optimistic, Pessimistic and Most Probable Demand for each product.
The next function, called `get_forecast(city, product)` returns the forecast over the next 8 years for any of the selected products in any city. The output is a DataFrame showing the optimistic, pessimistic and most probable demand as well as the associated revenues (taking into account the sales during Black Friday). This function will prove to be very useful in order to compute the `daily_forecast_simu()`.
`daily_forecast_simu(city, product)` is the last function of the Task 1.3. Similarly to the the previous ones, it takes into parameter the city, the product and generates a

scenario for the daily demand over the next 8 years. To model the demand share for each product, we decide to use a triangular distribution, since we are given a min, max and expected value (and then normalize so the sum of demand shares equals to 1). For the weekly and monthly shares, we will model them following a normal distribution, as stated in the wording, with a coefficient variation of .2.
The output are two data frames: `filtered_forecast` that returns the years demands, and `daily_demand_df` that returns a granular daily demand for a product over the 2026 - 2023 horizon. We can plot this demand over time and obtain the following result:

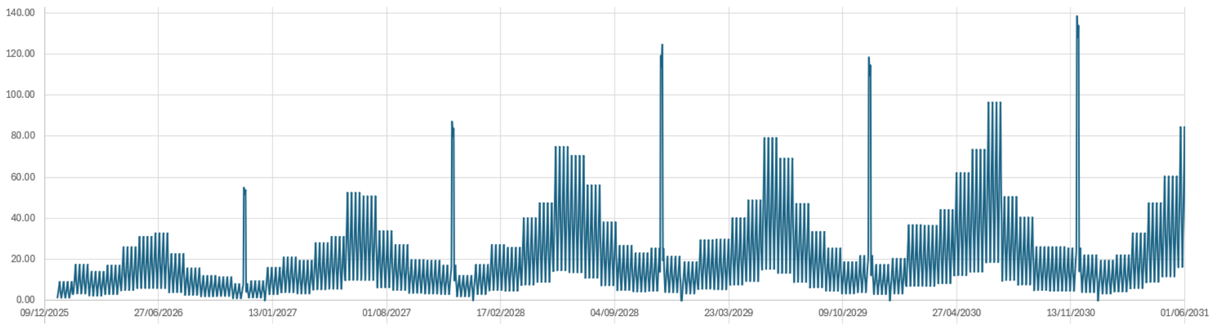


Figure 3: Daily Demand for Paris K10 - Optimistic Scenario

We can see peaks in demand, corresponding to Black Friday as well as a trend and a seasonality in the evolution of the demand.
Graphical User Interfaces (GUI) on Python (TKinter Library) helps the user better understand the dynamics of the demand. This can be computed fairly rapidly by using LLM models available on the Internet for free.

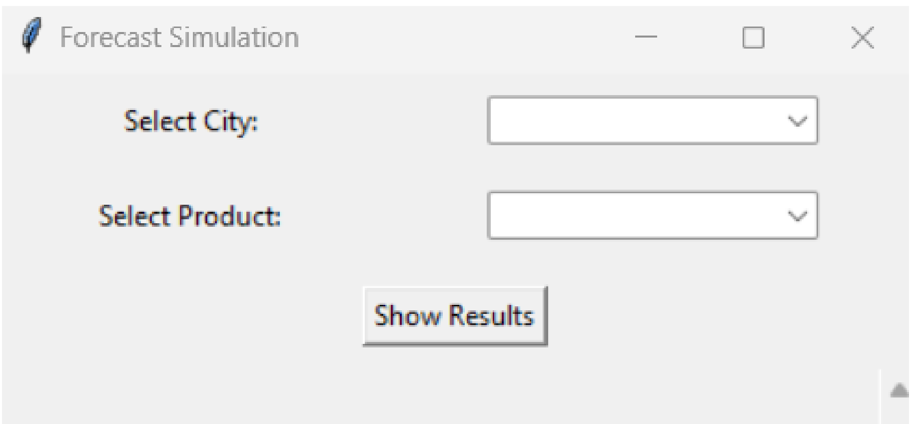


Figure 4: Graphical User Interface

For example, selecting *Lyon* and *S10* in the dropdown and clicking on *Show Results*:

Figure 5: Graphical User Interface Results

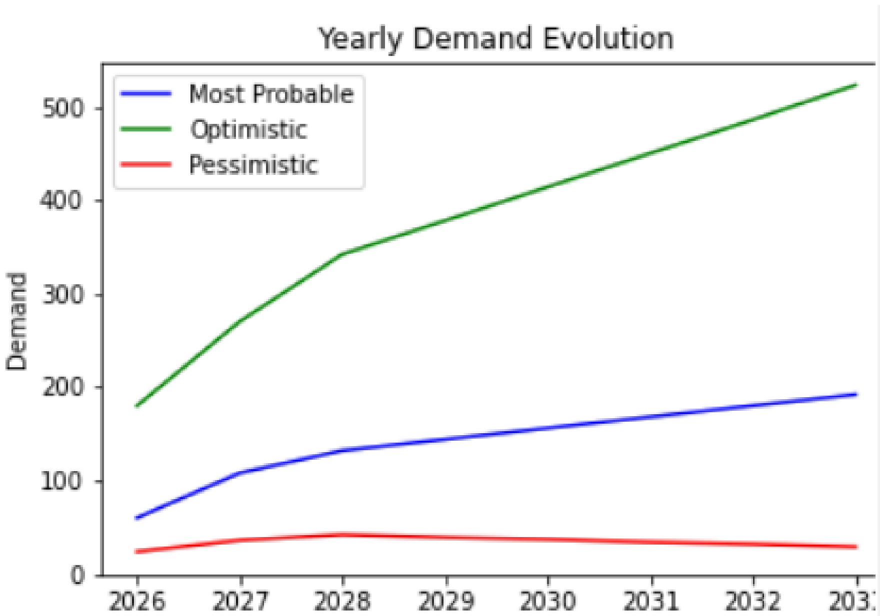


Figure 6: Demand Evolution - Lyon S10

Logically, the returns follow the yearly demand evolution.

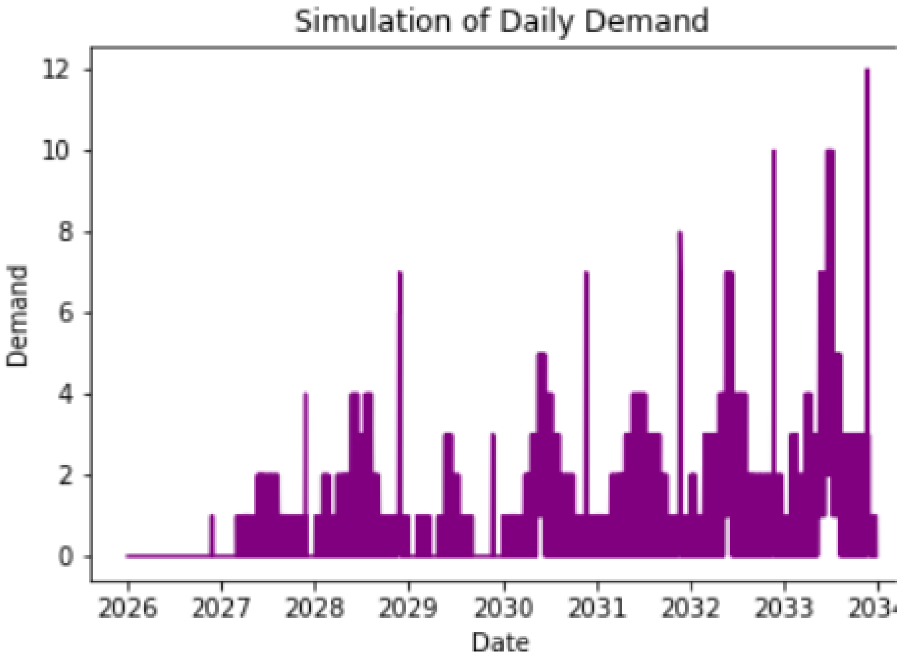


Figure 7: Daily Demand - Lyon S10

The code is available for review in the file **Python Visualization Script.py**.

1.4 Development of a daily demand simulator for BotWorld's expansion in Europe

For Task 1.4, we will be using the `daily_demand_df` function in order to generate daily scenarios. We will couple it with the `generate_scenarios(n, city, product)` function that creates n unique demand scenarios and plots the results using the `matplotlib.pyplot` library.

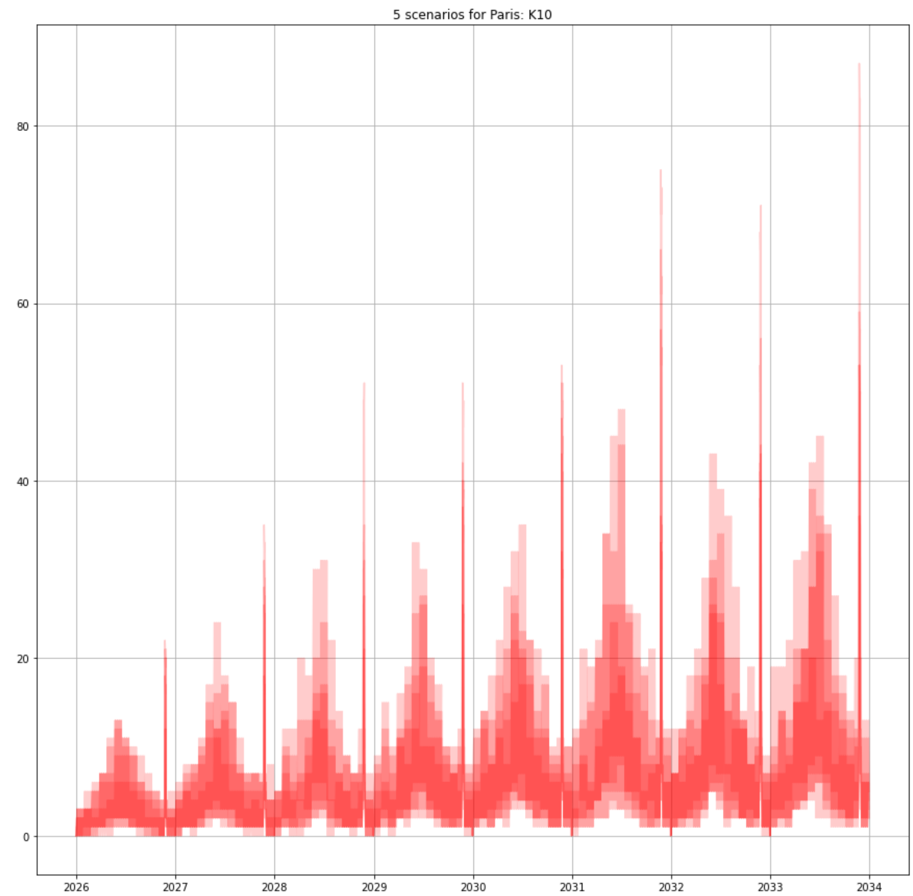


Figure 8: Monte Carlo Simulation with 5 iterations - Paris K10

The next plot provides a zoom into the first year of the previous simulation, with 15 iterations.

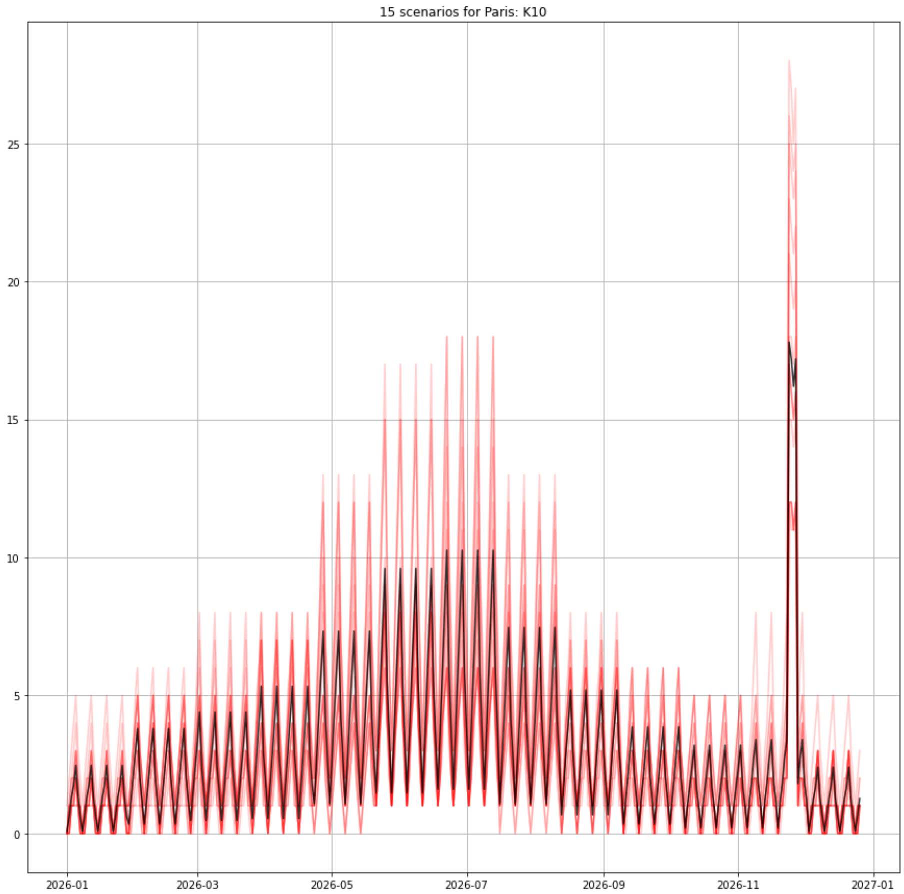


Figure 9: Monte Carlo Simulation with 15 iterations - Paris K10 - Y1

In the previous figure, the black curve represents the mean of all the simulations. An easy way to calculate the demand with a 99% confidence level is to use the 1st percentile of the highest simulated demand values. This ensures that there is a 99% probability that the actual demand will not exceed this value, providing a robust approach to planning for uncertainty.
However, in our case, this would require too many computation time. Therefore, we decided to simulate three scenarios of the *Optimistic Demand* and then calculate the mean. This should give a relatively good indicator for a 99% robust demand.
With 67 cities, 20 products and 3 simulations per tuple (city, product), this gives a number of:

$$67 \times 20 \times 3 \approx 4,000 \text{simulations}$$

Running the simulations takes around 2.5 hours, this relatively long running time is due to the important number of operations conducted on pandas DataFrames. Some methods can be used in order to improve this value (for example using Python Dictionaries

instead of DataFrames), however this code is written to only be ran once so taking the time to optimize it may not be worth it.
The results are shown in the attached file in the folder *Task 1 - Task 1.4*

1.5 Key facts and insights to BotWorld

The evolution of demand for BotWorld reveals some key insights. Notably, demand varies across different regions within a country. For instance, demand in capital cities is expected to rise significantly in the coming years, whereas in other cities within the same country, it is projected to remain relatively stable (see Figure 10 for France).

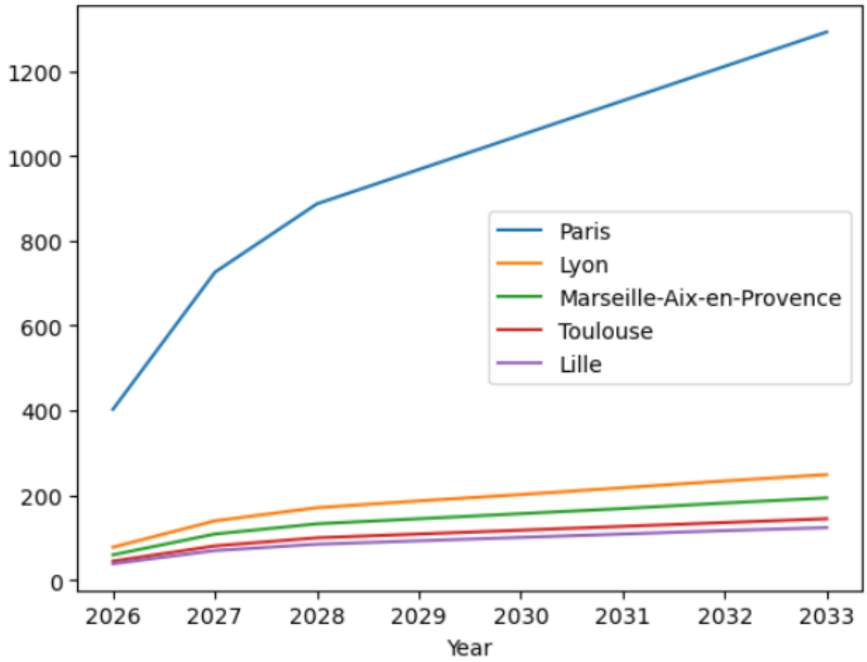


Figure 10: Evolution of Demand in France - Most probable - K10

This provides insight into how the population in Europe is expected to evolve in the coming years. The phenomenon of rural areas being overlooked in favor of larger cities is expected to intensify, with smaller urban areas experiencing significantly slower development compared to capital cities. Therefore, the projections of the population by the United Nations should be studied carefully by the company since the primary markets are likely to be in the major European Capitals.

Using data analytics, we can provide interactive maps that show the evolution of the demand for a specified product from 2026 to 2033.

Figure 11: 2026: Demand Across Europe - K10 - Most Probable

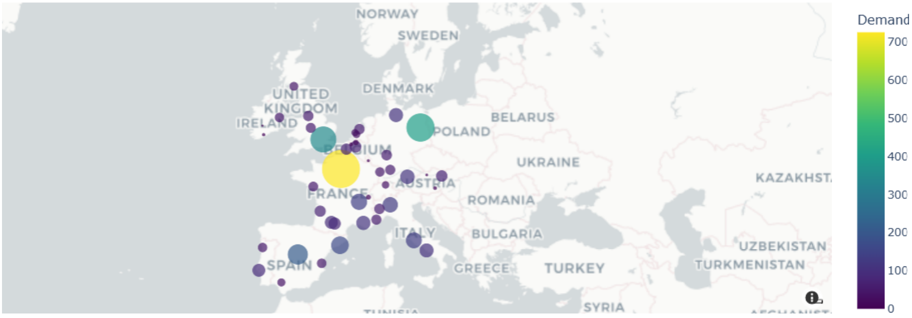


Figure 12: 2027: Demand Across Europe - K10 - Most Probable

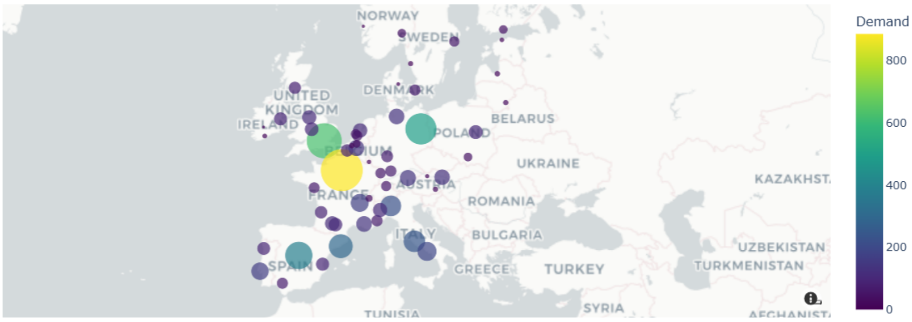


Figure 13: 2028: Demand Across Europe - K10 - Most Probable

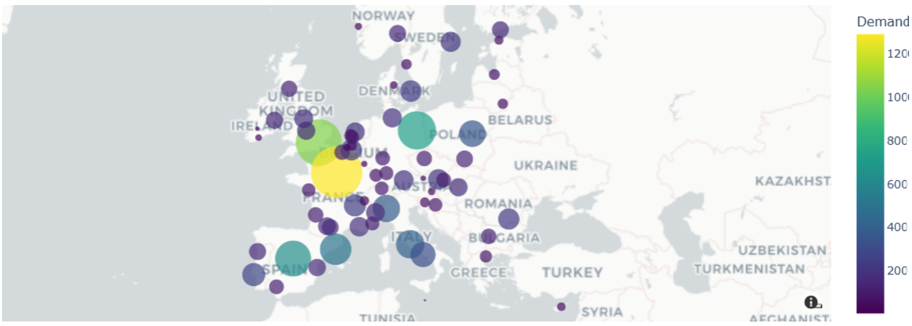


Figure 14: 2033: Evolution of the Demand Across Europe - K10 - Most Probable

The sequence of maps shows the evolution of the demand. We decided to provide the first 3 years (2026-2028) as well as the last year (2033). We can observe the market penetration for WorldBot, starting in France and Germany (primarily Paris and Berlin) and then spreading through Europe.
Other key insights include:

- Concentration in Capitals and Major Cities: consistent pattern of higher demand in capital cities and major urban centers like Paris, London, and Berlin. This reflects the trend of population and economic activity being concentrated in larger cities.
- Incremental Growth Over Time: The demand increases progressively each year, with noticeable growth in already high-demand areas. Capitals and large metropolitan regions are projected to experience the most significant increases.
- Stable or Marginal Growth in Smaller Areas: Smaller cities and rural areas exhibit relatively stable demand over time, with little to no significant increase compared to urban centers. This echoes the projected demand in Figure 11

These trends align with the phenomenon of urbanization, where resources and populations are increasingly concentrated in large cities. This provides a clear basis for the following market categorization:

1. **Primary Market:** large urban areas, especially capital cities
2. **Secondary Market:** smaller urban areas (non capital cities)
3. **Tertiary Market:** rural areas

2 TASK 2: Dedicated Network and Operations Design and Estimation

2.1 European fulfillment network with minimized number of fulfillment centers to meet adequate nearness

We will write an optimization model to determine where to locate the FCs and the DC in the network so as to minimize the travel distance. We will start by determining the road travel distances between all possible cities of the network.

2.1.1 Distance Matrix

We will use the Open Source Routing Machine (OSRM) to determine the distances between all cities of the network. The computations have been performed in the Python file `task2.1_distances`. The result is displayed in the csv file *driving_distances_matrix.csv*

2.1.2 Choice of fulfillment centers - 5 hours nearness

Sets and Indices

- I: Set of all cities (markets), indexed by i
- J: Set of potential FC locations (same as I), indexed by j
- M: Location of Metz DC (fixed)

Parameters

- d_{ij}: Distance between city i and city j (km)
- d_{Mj}: Distance between Metz DC and potential FC location j (km)
- D_{max}: Maximum allowable distance between market and FC (400 km assuming an average speed of 80 km/h and taking into account 5 hours adequate nearness)
- W: Large weight for the primary objective (10,000)

Decision Variables

- $y_j \in \{0, 1\}$: Binary variable indicating if an FC is located in city j

$$y_j = \begin{cases} 1 & \text{if an FC is located in city } j \\ 0 & \text{otherwise} \end{cases}$$

- $x_{ij} \in \{0, 1\}$: Binary variable indicating if market i is assigned to FC j

$$x_{ij} = \begin{cases} 1 & \text{if market } i \text{ is assigned to FC } j \\ 0 & \text{otherwise} \end{cases}$$

Objective Function

The objective function has two components with different priorities:

$$\min Z = W \sum_{j \in J} y_j + \sum_{j \in J} d_{Mj} y_j$$

Where:

- First term ($W \sum_{j \in J} y_j$): Minimizes the number of FCs
- Second term ($\sum_{j \in J} d_{Mj} y_j$): Minimizes total distance from Metz DC to FCs
- W is a large weight (10,000) to ensure priority of the first objective

Constraints

Single Assignment Constraint

Each market must be assigned to exactly one FC:

$$\sum_{j \in J} x_{ij} = 1 \quad \forall i \in I$$

FC Opening Constraint

Markets can only be assigned to locations where FCs are placed:

$$x_{ij} \leq y_j \quad \forall i \in I, j \in J$$

Maximum Distance Constraint
Markets cannot be assigned to FCs beyond the maximum distance:

$$x_{ij} = 0 \quad \forall i \in I, j \in J \text{ where } d_{ij} > D_{max}$$

This program has been coded using gurobi in the file `task2.1_FC_DC_400km`. The first output had connected Riga to Helsinki but this is considering that the container goes on a ferry, which we have decided to avoid so as not to add extra cost. Same for Birmingham connected to Dublin. So we will update the code to make sure the container doesn't have to go on the sea.

We need **23 FC** located in the following cities:

Bordeaux	Strasbourg	Hamburg	Vienna
Brussels	Dublin	Rome	Turin
Porto	Valencia	Seville	Manchester
Aarhus	Tallinn	Vilnius	Bergen
Krakow	Stockholm	Gothenburg	Sofia
Nicosia	Athens	Valletta	

Table 5: 23 Fulfillment Centers to open

All our markets are to be assigned to a FC according to the following table:

FC	Markets Served
Bordeaux	Toulouse, Nantes, Bordeaux
Strasbourg	Strasbourg, Munich, Frankfurt, Stuttgart, Zurich
Hamburg	Berlin, Hamburg
Vienna	Vienna, Graz, Linz, Zagreb, Prague, Budapest, Bratislava, Ljubljana
Brussels	Brussels, Paris, Lille, Cologne, Antwerp, Ghent, Luxembourg City, Amsterdam, Rotterdam, The Hague
Dublin	Dublin, Cork, Limerick
Rome	Rome, Naples
Turin	Lyon, Marseille, Nice, Milan, Turin, Geneva
Porto	Lisbon, Porto
Valencia	Madrid, Barcelona, Valencia
Seville	Seville

Manchester	London, Manchester, Birmingham, Glasgow
Aarhus	Copenhagen, Aarhus
Tallinn	Tallinn, Helsinki
Vilnius	Riga, Vilnius
Bergen	Bergen
Krakow	Warsaw, Krakow
Stockholm	Stockholm
Gothenburg	Oslo, Gothenburg
Sofia	Thessaloniki, Bucharest, Sofia
Nicosia	Nicosia
Athens	Athens
Valletta	Valletta

Table 6: Markets served by each FC

The FC are plotted on the following map:

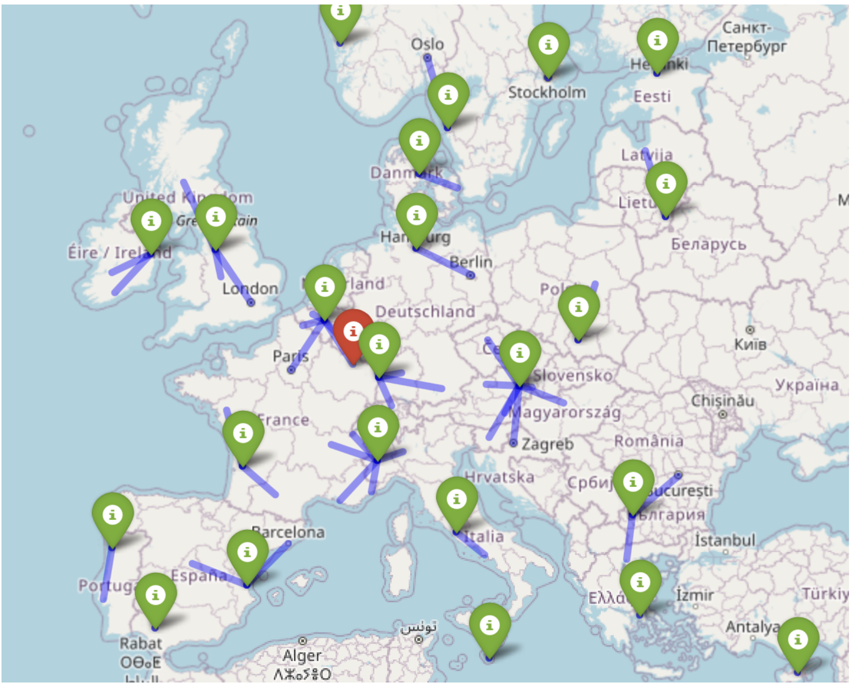


Figure 15: Map displaying the FC and their regions

. By running the code, you will get as an output an html link providing an interactive map on which you can zoom in and out.

2.1.3 Choice of fulfillment centers - 5 hours nearness vs 6 hours

If we were to consider 6 hours nearness (distance of 480km assuming 80 km/h average speed) instead of 5 hours, as suggested, we could get rid of 3 FC (Stockholm, Bergen, Seville) and only need **20** of them. It might make sense to consider this option too as it would save cost and we would still be able to cover a one day delivery as European drivers can drive up to 9 hours a day. We use the same code as before, changing only the distance limit to 480km in the new python file `task2.1_FC_DC_480km`. The 20 cities in which we should build the FCs are displayed in the following figure.

Lyon	Hamburg	Valencia	Vilnius
Bordeaux	Linz	Birmingham	Oslo
Metz	Dublin	Aarhus	Krakow
Gothenburg	Tallinn	Lisbon	Sofia
Rome	Nicosia	Athens	Valletta

Table 7: 20 Fulfillment centers to open when considering 6 hours nearness

The following figures show the difference between those two configurations. We note also that depending on the nearness chosen, the FC to build are not the same: in the first scenario, we have Lille and Strasbourg that are not part of the second one, being replaced by Metz for instance. this is due to the fact that we are trying to minimize travel distance from.to the DC in priority.

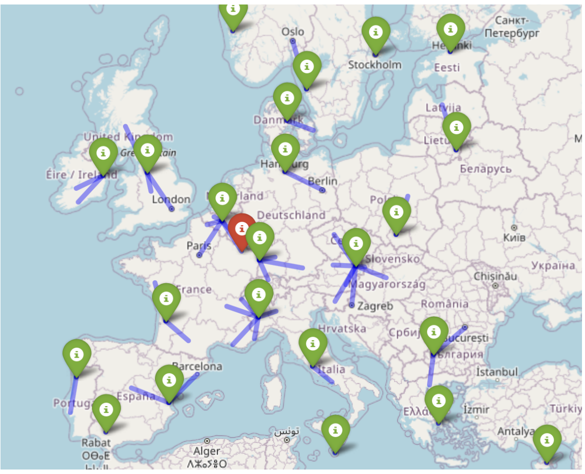


Figure 16: 23 FC - 5 hours nearness

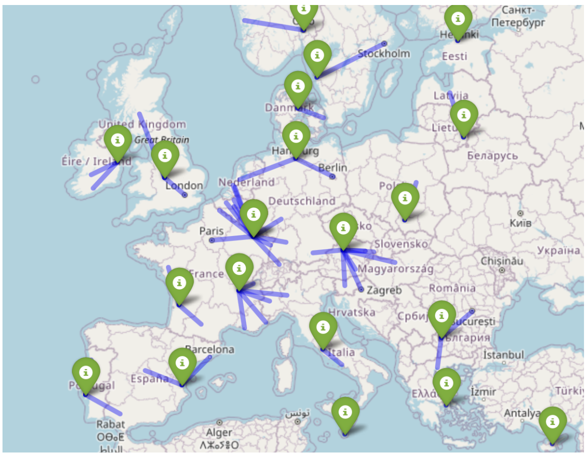


Figure 17: 20 FC - 6 hours nearness

Nevertheless, this casework wants us to focus on a 5 hours nearness and since the maximum speed of a truck in Europe is 90km/h, the actual maximum distance he could cover in 5 hours is 450km. Since a truck doesn't always drive at its maximum speed, the is not reachable and we will focus on having 23 FC as stated in table 5.

2.2 Estimation of each FC's capacity

We first collect data on the number of pallets required at each FC each year derived from data from Task 1.4 and Task 2.1. Based on the pallet requirements, we will write an optimization model to categorize each FC into tiers {Small, Medium, Large} each year. We will also collect data on the maximum throughput of trucks and pallets at each FC each year to estimate the number of dock doors and the number of workers required at each tier of FC each year.

2.2.1 Storage Capacity Estimation

Based on data from Task 1.4 and Task 2.1, we attribute each market to each FC and calculate the required storage capacity for each product at each FC each year. Combining the given data of product boxing sizes and two-high pallet sizes, we calculate the maximum number of pallets required at each FC each year to store maximums of all products in model-specific pallets. In doing so, we only use the perimeter pallet (the one with a smaller width) when the two can take the same number of boxes. Otherwise, we use the wooden pallet (the one with a larger width). The results are shown in the attached file *:/Task 2/Task 2.2/Storage Capacity.csv*

2.2.2 Throughput Estimation

We attribute each market to each FC with a similar logic and calculate the throughput in mixed-product pallets and trucks, including incoming products from the DC and outgoing destination-specific products.

In estimating the number of mixed-product pallets, we consider a combination of vacancy and volumetric analysis (total volume of pallets vs. total volume of the products) with a vacancy factor of 1.2. This means the required volume of pallets is 1.2 times the total volume of products, and the vacancy factor efficiently captures the vacant spaces that would appear when loading pallets.

We used the pallet throughput estimation to estimate the truck throughput. We assume that FC-to-market trips use **light rigid trucks** with capacities of 14 pallets, and the DC-to-FC trips use **heavy rigid trucks** with capacities of 20 pallets. These capacities are based on the online resources we gathered. Based on the given guideline, outgoing trucks are destination-specific.

The results are shown in the attached file *:/Task 2/Task 2.2/Throughput.csv*

2.2.3 Yearly Tier Assignment

We write an optimization model to categorize each FC into tiers {Small, Medium, Large} each year.

Parameters and Sets

- F: Set of FCs based on task 2,1 denoted by $f \in F$
- T: Set of tiers denoted by $t \in T$, with 1 = Small, 2 = Medium, and 3 = Large
- E: Set of years denoted by $e \in E$
- s_{fe}: storage capacity of FC f in year e derived from the previous subsection

Decision Variables

- $x_{fte} \in \{0, 1\}$: Whether FC f is tier t in year e

$$x_f te = \begin{cases} 1 & \text{if FC } f \text{ is tier } t \text{ in year } e \\ 0 & \text{otherwise} \end{cases}$$

- $y_{te} \in \mathbb{Z}$: The number of total pallets assigned for FCs in tier t in year e

Objective Function

The following is our objective function:

$$\min Z = \sum_{f \in F, t \in T, e \in E} y_{te} x_{fte} - x_{fte} s_{fe}$$

, where we minimize the total wasted storage capacity in each tier of FCs each year.

Constraints

Single Assignment Constraint

Each FC must have a tier each year:

$$\sum_{t \in T} x_{fte} = 1 \quad \forall f \in F, e \in E$$

Tier Order Constraint

Tier 1 (Small) has less capacity than Tier 2 (Medium), and Tier 2 (Medium) has less capacity than Tier 3 (Large):

$$y_{te} \leq y_{t+1,e} \quad \forall t \in \{1, 2\}, e \in E$$

Tier Assignment Constraint

FCs in each tier has smaller or equal storage capacity than the assigned tier:

$$x_{fte}s_{fe} \leq y_{te} \quad \forall f \in F, t \in T, e \in E$$

This program has been coded using gurobi in the file *:/Task 2/Task 2.2/ Task 2.2.html*. We realize that this model is nonlinear. However, since the problem size is small, the running time is fast, and there is no need for a linear model. The tier assignment results are shown in the attached file *:/Task 2/Task 2.2/FC Plans.csv*.

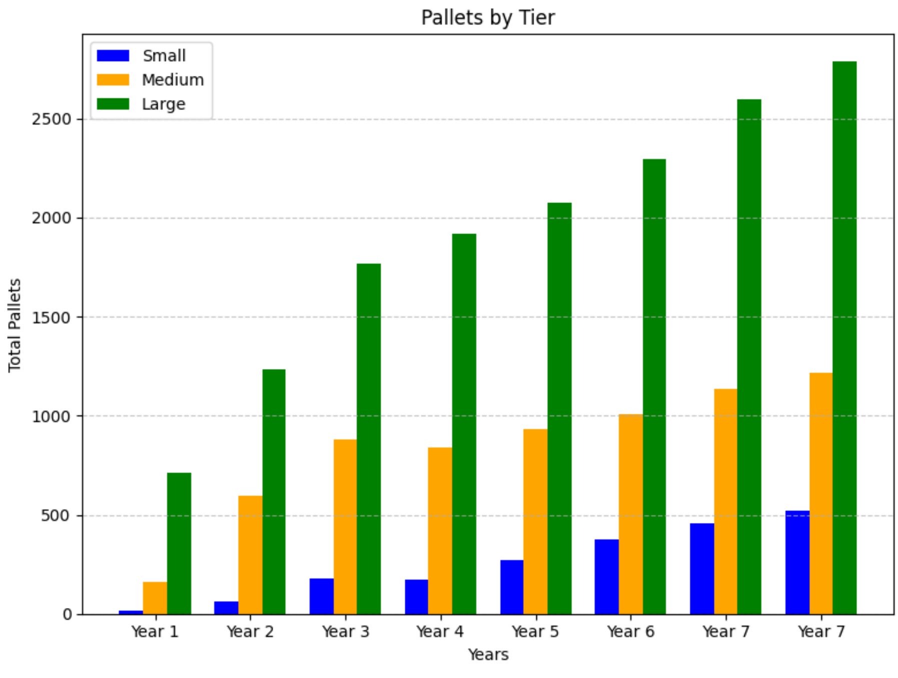


Figure 18: Pallets of Each Tier of FCs by Years

Figure 18 shows the optimization result of pallets assigned to each tier of FCs by years. Using this result, we calculate the number of 3-height racks for each tier of FCs by years shown in Figure 19. As the figure shows, the number of racks required at each tier of FCs increases as the year increases.

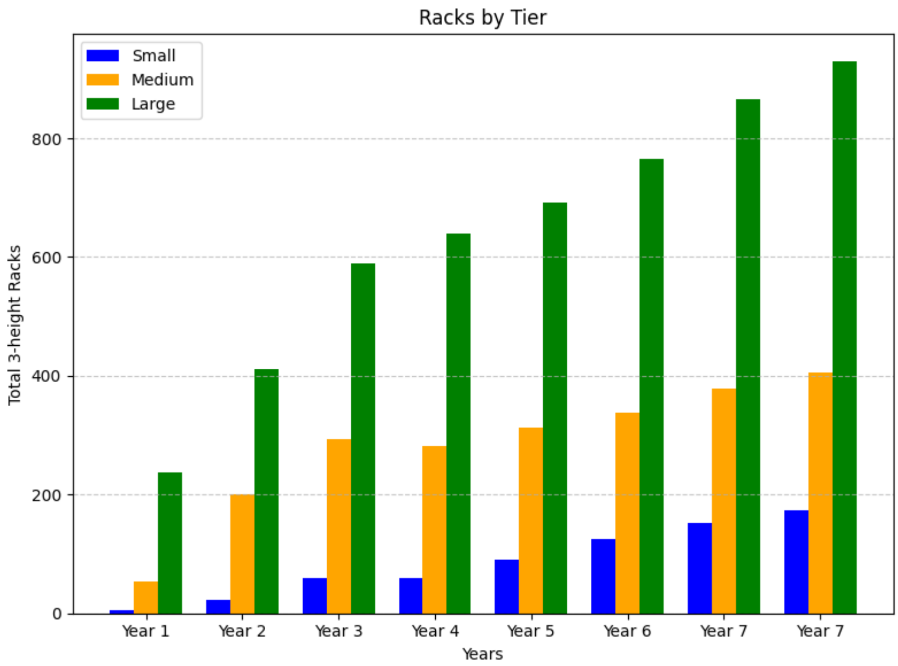


Figure 19: 3-Height Racks of Each Tier of FCs by Years

2.2.4 FC Tier Dock Door Assignment

In estimating the number of dock doors of each tier of FCs each year, we combine the truck throughput data and the tier plan data, and we assume:

- The FCs operate 12 hours a day.
- Incoming and outgoing trucks share the same set of dock doors.
- It takes one hour to load and unload a truck. Thus, a dock door can handle 12 trucks per day at best.

These assumptions lead us to calculate the number of dock doors required at each tier of FCs each year in order to accommodate the truck throughputs. The results are shown in the attached file *:/Task 2/Task 2.2/Tier Plans.csv*. Figure 20 visualizes the result. We want to point out that some tiers of FC require fewer dock doors as the year increases. This is because our generated simulation data on daily demand has some variations and is not strictly mono-increasing by year.

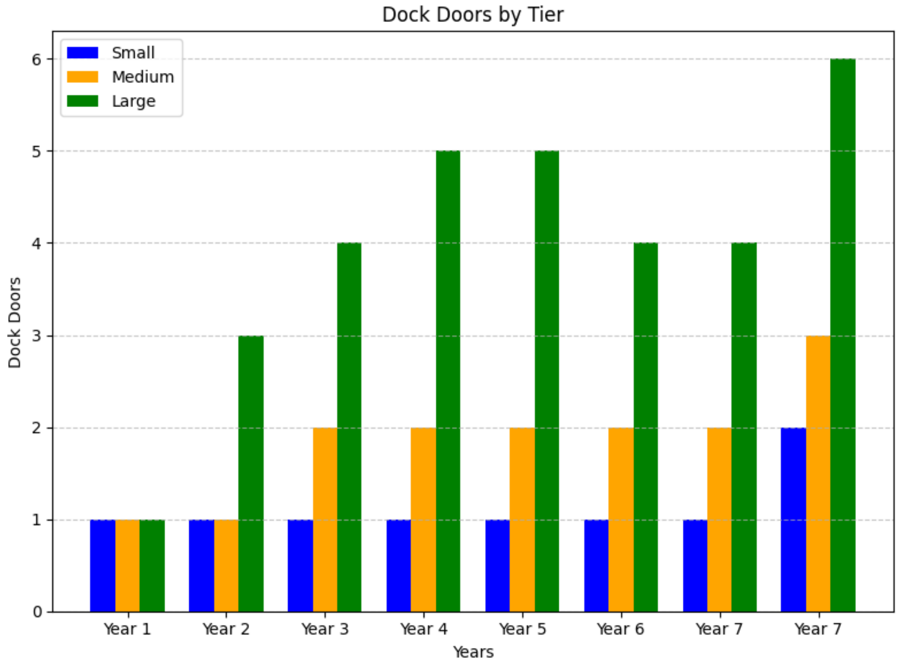


Figure 20: Dock Doors of Each Tier of FCs by Years

2.2.5 FC Tier Worker Assignment

In estimating the number of workers of each tier of FCs each year, we combine the pallet throughput data and the tier plan data, and we assume:

- The FCs operate 12 hours a day.
- Each rack takes $1.5m \times 1.5m$ spaces, which is large than the pallet size of $1.2m \times 1.2m$ accounting for walking and loading spaces.
- The overall size of each FC is $1.5\times$ the size of its racks, accounting for docking, resting, and working spaces.
- The docks surround the racks in the shape of a square. Thus, the average walking distance between dock doors to the racks is half of the side of the FC, and since the worker will need to travel back, the average walking distance for managing a pallet is the side of the FC.
- A worker can only handle one pallet at a time, and it takes 15 minutes for each worker to locate/load/organize a pallet.
- Together, the average time it takes for a worker to finish all processes associated with a pallet is $\sqrt{\lceil (y_{te}/3) \rceil * 1.5 * 1.5 * 1.5}/82 + 15$ minutes.

- So a worker can handle is $\lfloor 8 * 60/(\sqrt{\lceil (y_{te}/3) \rceil * 1.5 * 1.5 * 1.5}/82 + 15) \rfloor$ pallets per day on average

These assumptions lead us to calculate the number of workers required at each tier of FCs each year in order to accommodate the pallet throughputs. The results are shown in the attached file *:/Task 2/Task 2.2/Tier Plans.csv*. Figure 24 visualizes the result. We want to point out that some tiers of FC require fewer workers as the year increases. This is because our generated simulation data on daily demand has some variations and is not strictly mono-increasing by year.

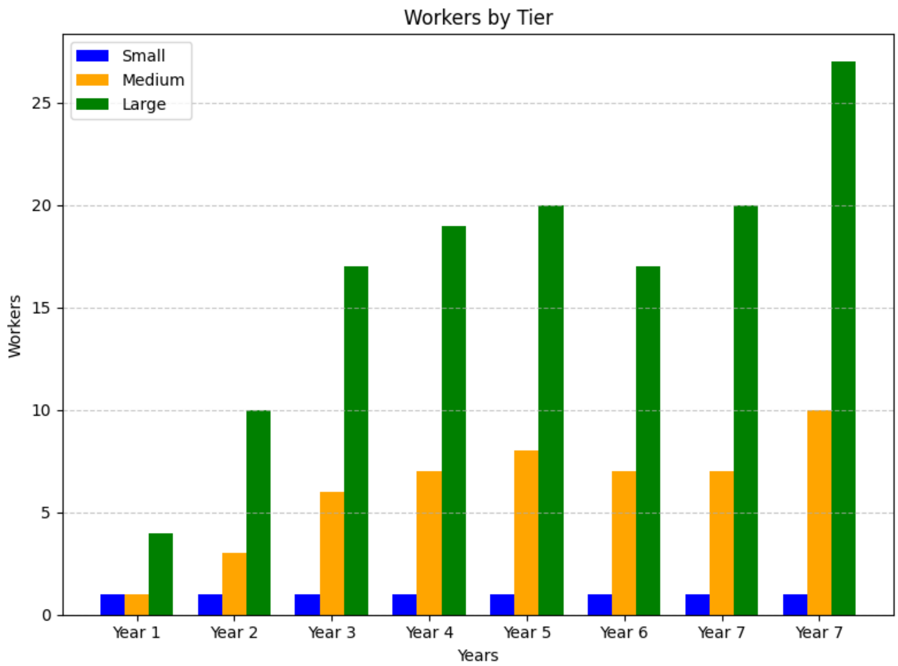


Figure 21: Workers of Each Tier of FCs by Years

2.3 Estimation of the European DC's required capacity

We first collect data on the number of pallets required at the DC each year derived from data from Task 1.4 and Task 2.1. We also collect data on the maximum throughput of trucks and pallets at the DC each year to estimate the number of dock doors and the number of workers required each year.

2.3.1 Storage Capacity Estimation

Based on data from Task 1.4 and Task 2.1, we calculate the required storage capacity for each product at the DC each year. Combining the given data of product boxing sizes and two-high pallet sizes, we calculate the maximum number of pallets required at the DC each year to store maximums of all products in model-specific pallets. In doing so, we only use the perimeter pallet (the one with a smaller width) when the two can take the same number of boxes. Otherwise, we use the wooden pallet (the one with a larger width). The results are shown in the attached file *:/Task 2/Task 2.3/DC Storage Capacity.csv*

2.3.2 Throughput Estimation

We calculate the throughput in mixed-product pallets and trucks, including incoming products from the supplier and outgoing FC-specific products.

In estimating the number of mixed-product pallets, we consider a combination of vacancy and volumetric analysis (total volume of pallets vs. total volume of the products) with a vacancy factor of 1.2. This means the required volume of pallets is 1.2 times the total volume of products, and the vacancy factor efficiently captures the vacant spaces that would appear when loading pallets.

We used the pallet throughput estimation to estimate the truck throughput. We assume that the DC-to-FC trips use **heavy rigid trucks** with capacities of 20 pallets, and the warehouse/assembly center-to-DC trips use **semi-trailer trucks** with capacities of 30 pallets. These capacities are based on the online resources we gathered. Based on the given guideline, outgoing trucks are FC-specific.

The results are shown in the attached file *:/Task 2/Task 2.3/DC Throughput.csv*

2.3.3 DC Rack Assignment

We calculate the number of 8-height racks for the DC by years shown in Figure 22. As the figure shows, the number of racks required at the DC increases as the year increases.

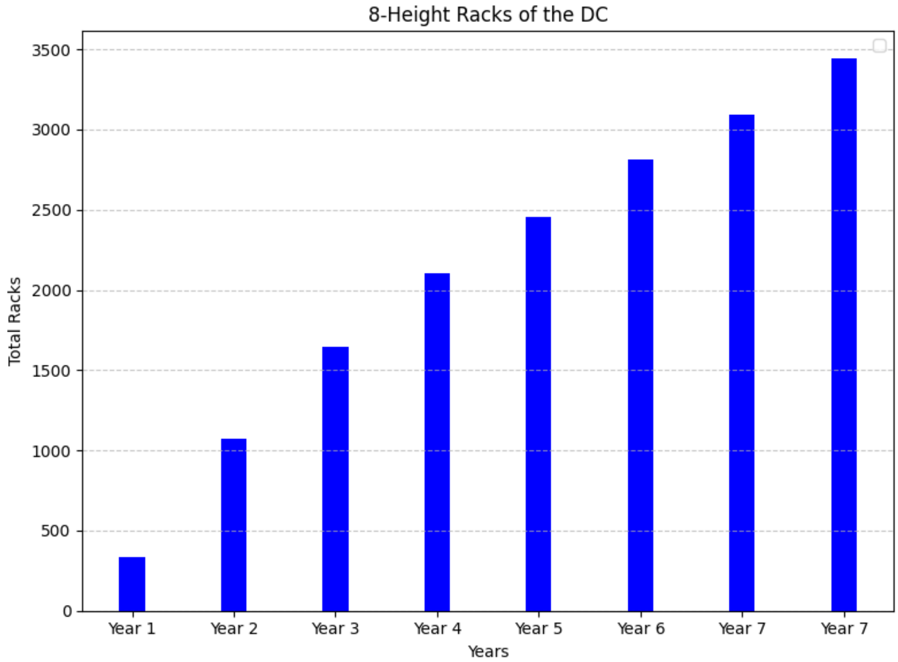


Figure 22: 8-Height Racks of the DC by Years

2.3.4 DC Dock Door Assignment

In estimating the number of dock doors of the DC each year, we combine the truck throughput data and the tier plan data, and we assume:

- The DC operates 12 hours a day.
- Incoming and outgoing trucks share the same set of dock doors.
- It takes one hour to load and unload a truck. Thus, a dock door can handle 12 trucks per day at best.

These assumptions lead us to calculate the number of dock doors required at the DC each year in order to accommodate the truck throughputs. The results are shown in the attached file *:/Task 2/Task 2.3/DC Plans.csv*. Figure 23 visualizes the result. We want to point out that sometimes, the DC requires fewer dock doors as the year increases. This is because our generated simulation data on daily demand has some variations and is not strictly mono-increasing by year.

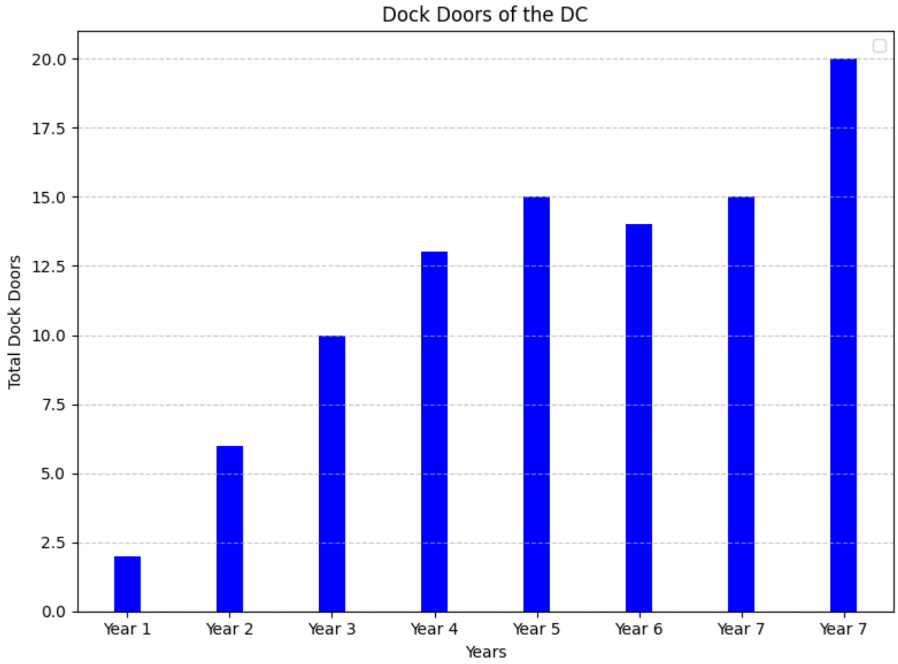


Figure 23: Dock Doors of the DC by Years

2.3.5 DC Worker Assignment

In estimating the number of workers of the DC each year, we combine the pallet throughput data and the tier plan data, and we assume:

- The DC operates 12 hours a day.
- Each rack takes $1.5m \times 1.5m$ spaces, which is large than the pallet size of $1.2m \times 1.2m$ accounting for walking and loading spaces.
- The overall size of the DC is $1.5\times$ the size of its racks, accounting for docking, resting, and working spaces.
- The docks surround the racks in the shape of a square. Thus, the average walking distance between dock doors to the racks is half of the side of the DC, and since the worker will need to travel back, the average walking distance for managing a pallet is the side of the DC.
- A worker can only handle one pallet at a time, and it takes 15 minutes for each worker to locate/load/organize a pallet.
- Together, the average time it takes for a worker to finish all processes associated with a pallet is $\sqrt{\lceil (max\_pallets/8) \rceil * 1.5 * 1.5 * 1.5}/82 + 15$ minutes.

- So a worker can handle is $\lfloor 12 * 60/(\sqrt{\lceil(max\_pallets/8)\rceil * 1.5 * 1.5 * 1.5}/82 + 15)\rfloor$ pallets per day on average

These assumptions lead us to calculate the number of workers required at the DC each year in order to accommodate the pallet throughputs. The results are shown in the attached file *:/Task 2/Task 2.3/DC Plans.csv*. Figure 24 visualizes the result. We want to point out that sometimes, the DC requires fewer workers as the year increases. This is because our generated simulation data on daily demand has some variations and is not strictly mono-increasing by year.

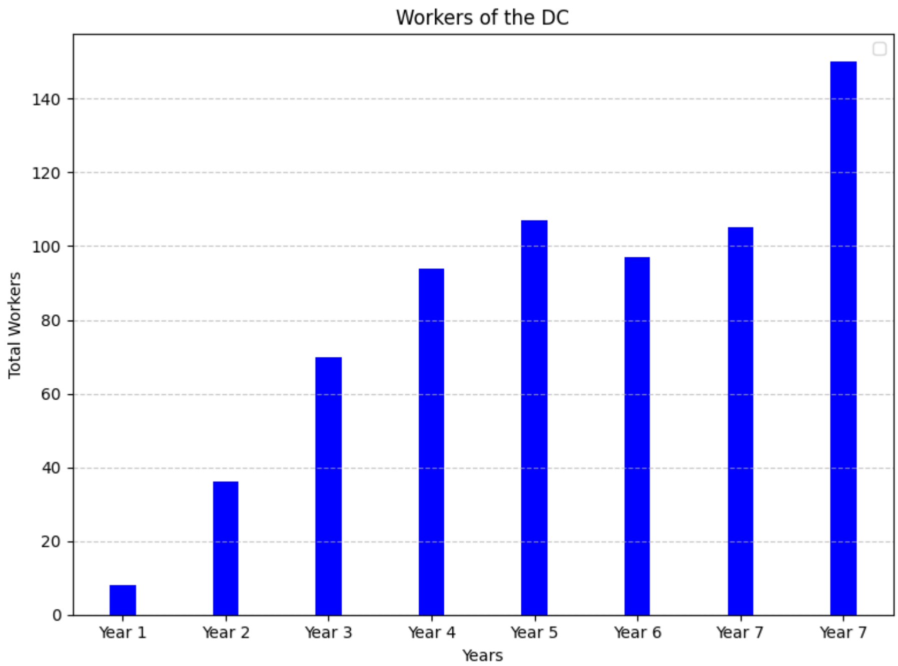


Figure 24: Workers of the DC by Years

2.3.6 Final Layout for the Distribution Center

The final layouts for the distribution center across the years are provided below (figure 25 and figure 26). It has the appropriate amount of dock doors and 8-racks as the data calculated in the previous subsections.

We also added basic accommodation and workplaces to add to the realism of this case study. The different color racks represent the additional racks that are to be added each year.

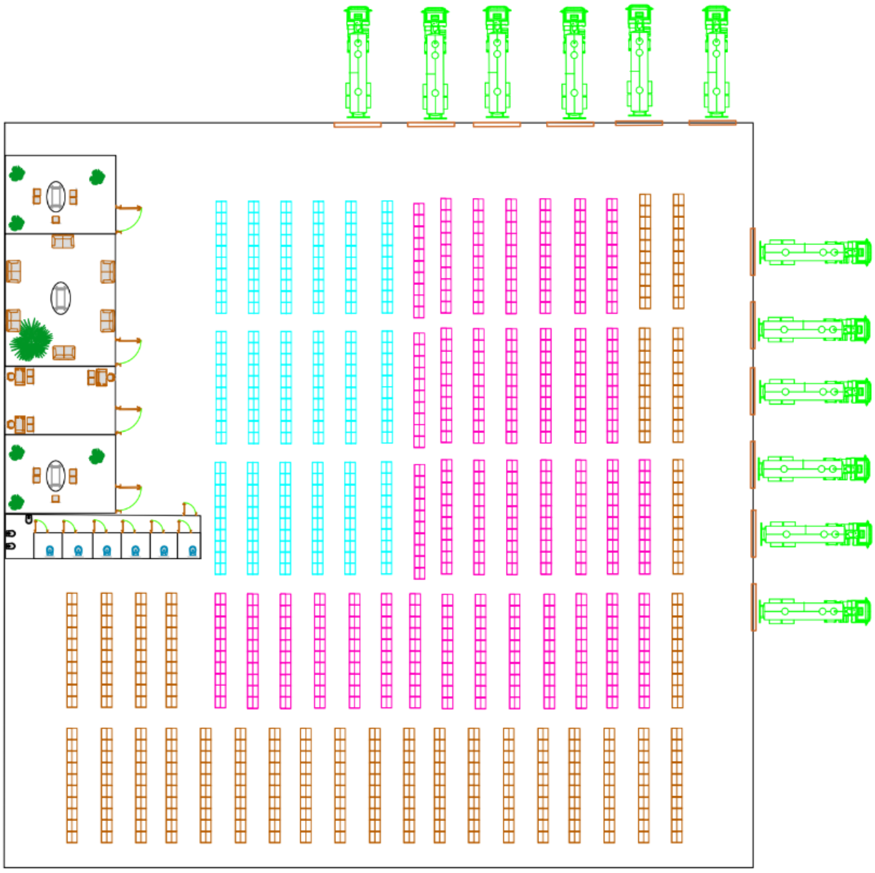


Figure 25: DC layout for 2026 - 2028

- Cyan: 8-racks for 2026
- Pink: 2027
- Brown: 2028

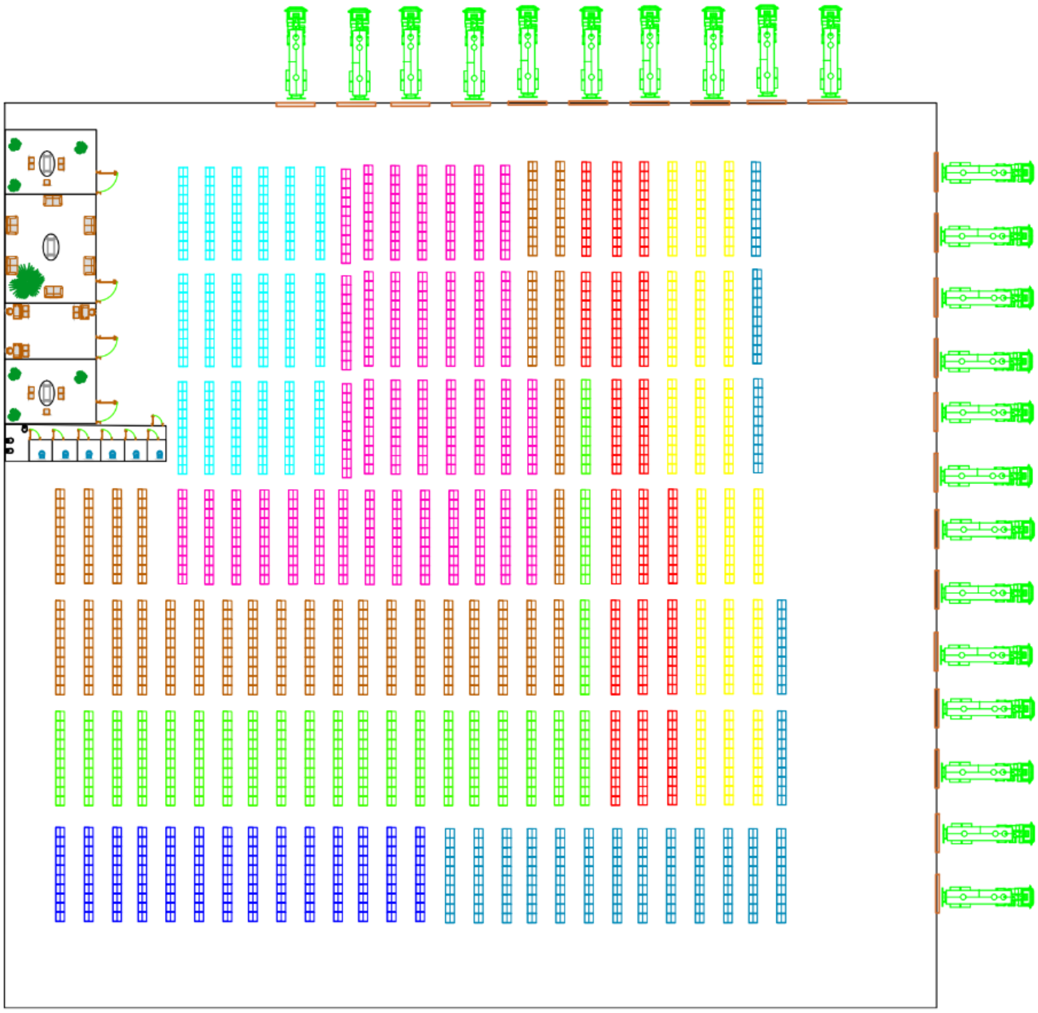


Figure 26: DC layout for 2029 - 2033

- Green: 2029
- Red: 2030
- Yellow: 2031
- Blue: 2032
- Dark Blue: 2033

2.4 Estimation of the required capacity of the factory to be built

In this task, we will use the output from task 1.4, `simulation_results.csv` displaying demand of each product in each market for each day of the studied period.

2.4.1 Yearly and periodically demand

We will aggregate the demand for each product for each year so as to size the facility accordingly every year. All computations have be done in the file *:/Task 2/Task 2.4/task2.4.py* and the output is stored in the file *:/Task 2/Task 2.4/task2.4_output.xlsx.*

For each year, we have the following demand:

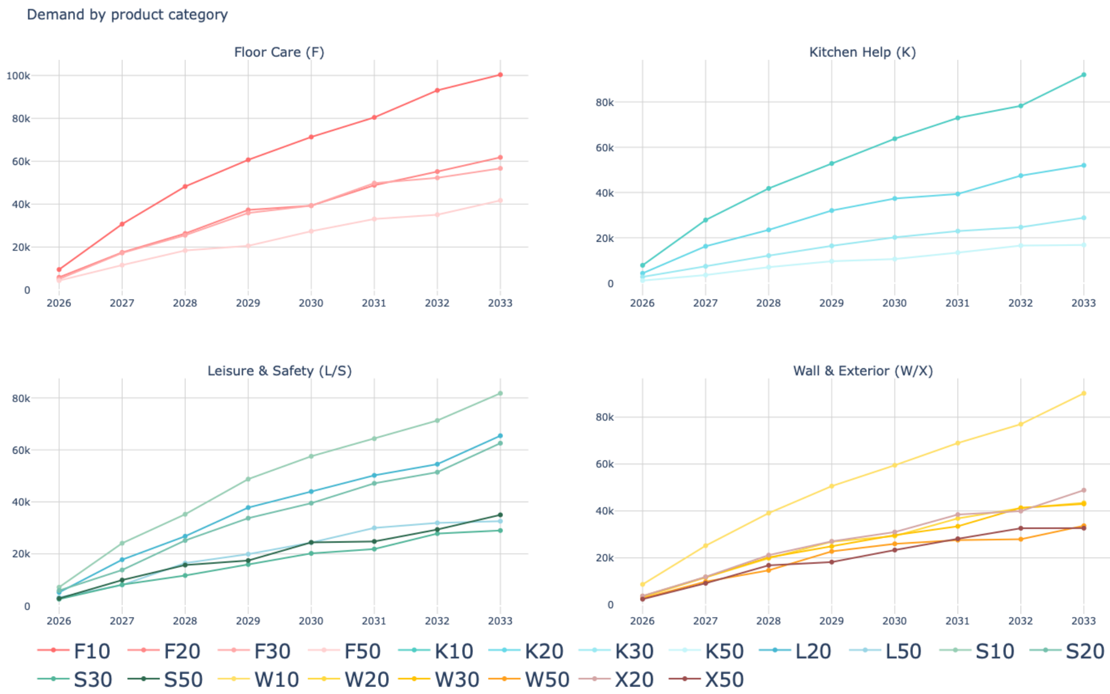


Figure 27: Yearly demand for each product

We observe that the demand is growing for each product over the years as expected.

We can also get the demand for each 13 period of each year. The result is displayed in figure 28. We will use this result to compute the stock to be expected at the facility during each period

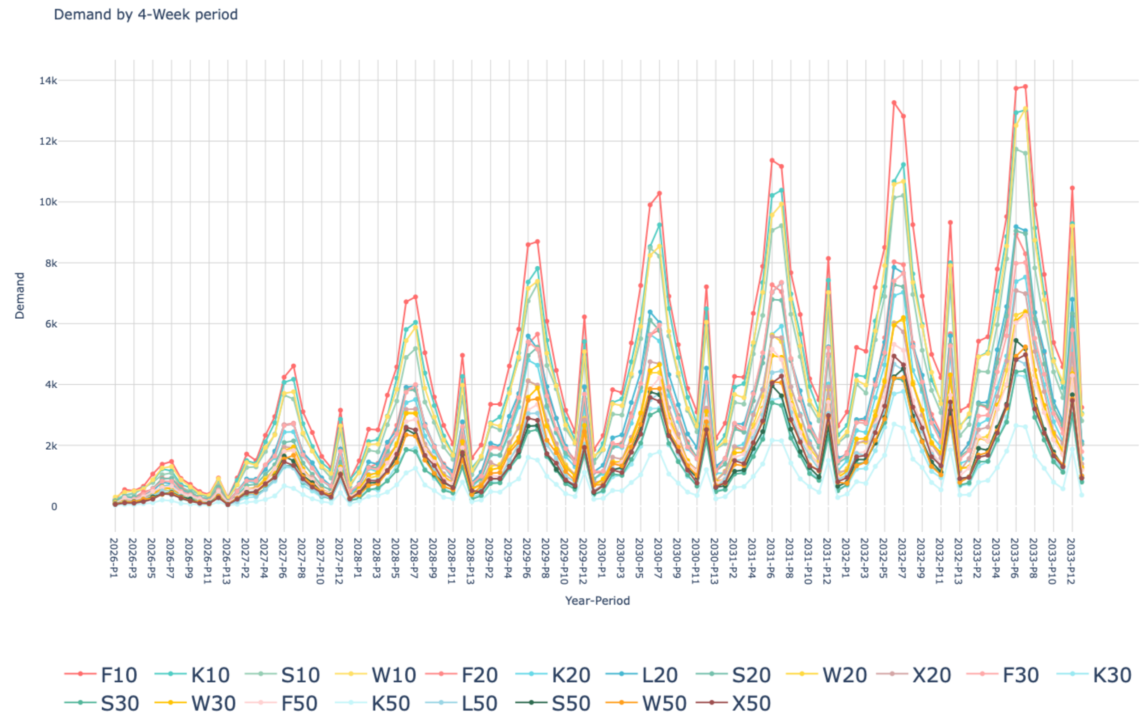


Figure 28: Periodic demand for each product from 2026 to 2033

2.4.2 Stock levels

Considering a 4 week safety stock for Chinese supplies and 2 weeks for European, and 11 days of French Holidays, we will compute the stock levels:

- The weighted safety stock is computed as:

$$\text{Weighted safety stock} = (0.6 \times 2 + 0.4 \times 4) \text{ weeks} = 2.8 \text{ weeks}$$

- Since each period is 4 weeks, the safety stock factor applied to demand is:

$$1 + \frac{2.8}{4} = 1.7$$

- France has 11 public holidays per year, distributed across 13 periods. The working days per period are:

$$\text{Working days per period} = 28 - \frac{11}{13} \approx 27.15$$

The holiday adjustment factor is:

$$\text{Holiday factor} = \frac{28}{\text{Working Days per Period}} = \frac{28}{27.15} \approx 1.031$$

- The required stock level for each period is:

$$\text{Stock level} = \text{Period Demand} \times 1.7 \times 1.031$$

The results are plotted in figures 29 and 30.

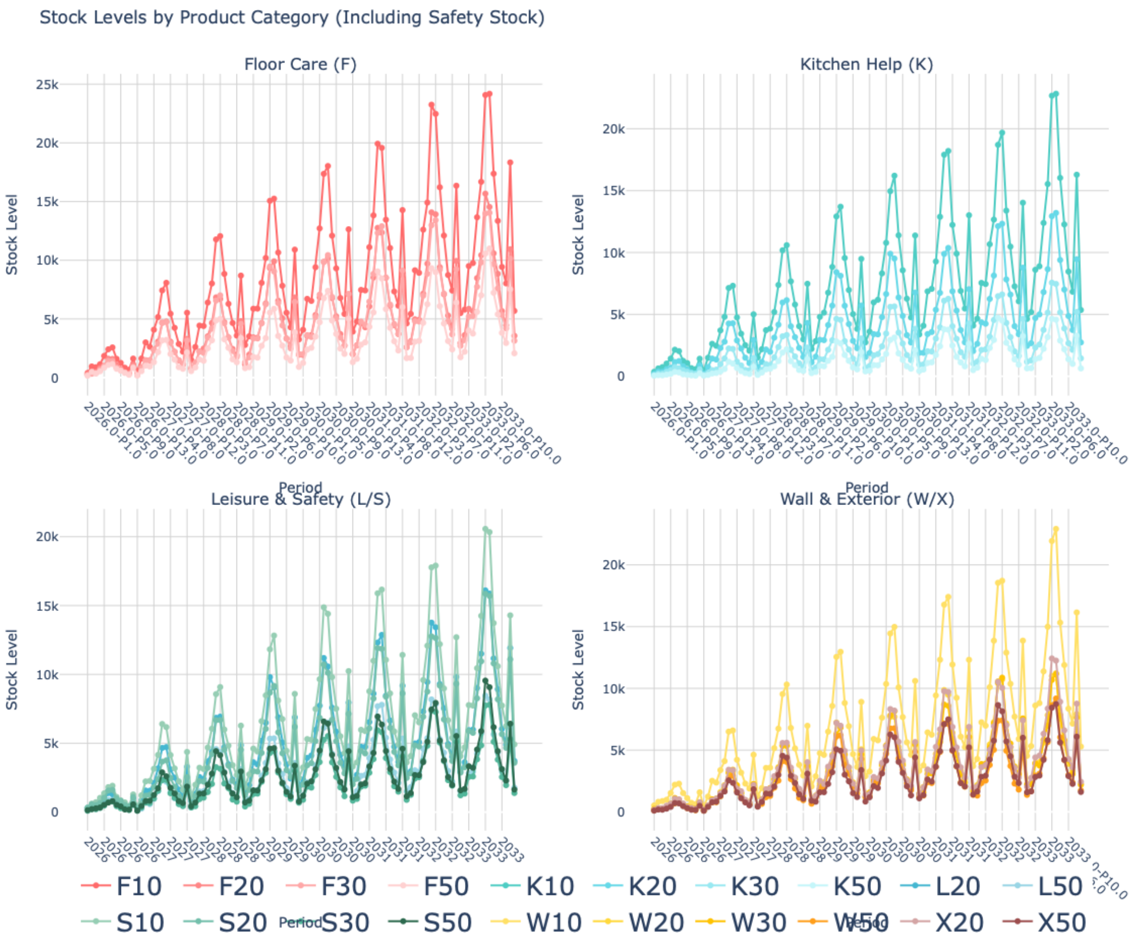


Figure 29: Periodic stock for each product from 2026 to 2033

The graph is divided into four subplots, each representing a different product category:

- **Floor Care (F)** (Top-left, red shades)

- **Kitchen Help (K)** (Top-right, blue shades)
- **Leisure & Safety (L/S)** (Bottom-left, green shades)
- **Wall & Exterior (W/X)** (Bottom-right, yellow/brown shades)

Each line corresponds to a specific product model, showing stock levels over time. The trend indicates periodic fluctuations, with noticeable peaks occurring at regular intervals, likely due to demand seasonality and replenishment cycles.

- Stock levels exhibit sharp peaks, likely aligning with high-demand periods or bulk supply shipments.
- **Floor Care (F)** and **Kitchen Help (K)** categories show **higher stock volatility**, indicating stronger demand fluctuations.
- **Leisure & Safety (L/S)** and **Wall & Exterior (W/X)** categories have **comparatively lower peaks**, suggesting more stable or lower demand.

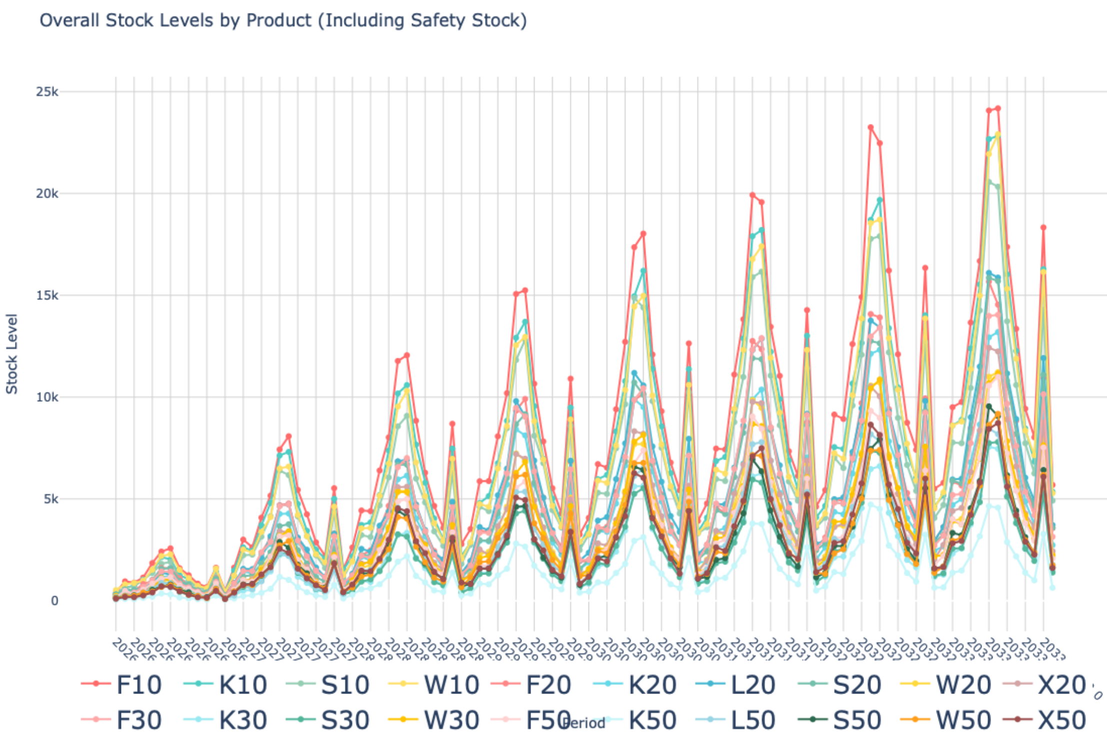


Figure 30: Yearly stock for each product from 2026 to 2033

This graph aggregates all product categories into a **single plot**, allowing for an overall trend analysis. The cyclical pattern is evident, showing recurring spikes in stock levels at consistent intervals, likely due to periodic replenishment and demand surges.

- **Stock levels across all products peak synchronously**, which may indicate synchronized production and inventory policies.
- Some products, like **F10 and K10**, seem to have the highest peaks, implying either high demand or a larger safety stock allocation.

2.5 Optimized combination of assembly lines

Based on Task 2.4's yearly production requirement of each product in the factory, we propose an optimization model to determine the yearly optimal assembly line combinations in order to accommodate the production requirements. In doing so, we assume that each type of assembly line's capacities dedicated to each product line are fixed between different lines of the same. We solve the optimization model separately each year and document the outputs.

Parameters and Sets

- P: Set of products denoted by $p \in P$
- A: Set of assembly lines denoted by $a \in A$, with $a \in \{1, 2, 3, 4, 5, 6\}$ referring to respective single-category assembly lines of {Floor Care (F), Kitchen Help (K), Leisure (L), Safety & Security (S), Wall & Window Care (W), and Exterior Care (X)}, and $a \in \{7, 8, 9\}$ referring to respective multi-category assembly lines of {F + K + L, S + W + X, All}
- $C(a)$: Set of products assembly line a is capable of assemble, $C(a) \subseteq P$
- r_p: Requirement of product p in the current year
- c_a: Yearly cost of assembly line a

Decision Variables

- $x_a \in \mathbb{Z}$: The number of assembly line a in the current year
- $y_{a,p} \in \mathbb{Z}$: The amount of capacity of assembly line a allocated to product p

Objective Function

The following is our objective function:

$$\min Z = \sum_{a \in A} x_a c_a$$

, where we minimize the total cost of the assembly lines of the current year.

Constraints

Product Requirement Constraint

Each product's requirement must be satisfied:

$$\sum_{a \in A} x_a y_{a,p} \geq r_p \quad \forall p \in P$$

Assembly Line Capacity Constraint

Each assembly line's capacity is less than 100:

$$\sum_{p \in P} y_{a,p} \leq 100 \quad \forall a \in A$$

Assembly Line Capability Constraint

Each assembly line is only capable of producing the defined products:

$$y_{a,p} = 0 \quad \forall a \in A, p \notin C(a)$$

This program has been coded using gurobi in the file *:/Task 2/Task 2.5/Task 2.5.html*. We realize that this model is nonlinear. However, since the problem size is small, the running time is fast, and there is no need for a linear model. Figure 31 shows the yearly cost (derived from the given daily operational cost of each assembly line) based on our optimization model. The total cost increases as the year increases. The planning results are shown in the attached folder *:/Task 2/Task 2.4/Yearly Plan.csv*. Figure 32 demonstrates the assembly line counts by years. As one can observe, the yearly patterns are similar between the types of assembly lines, which is because of the similarity in yearly demand patterns we derived from Task 2.4. It is interesting to see that assembly lines capable of multiple products are rarely picked. This might be due to their higher costs. Then number of different types of assembly lines our optimization model proposes are all increasing by years, which corresponds to the yearly increased demand data.

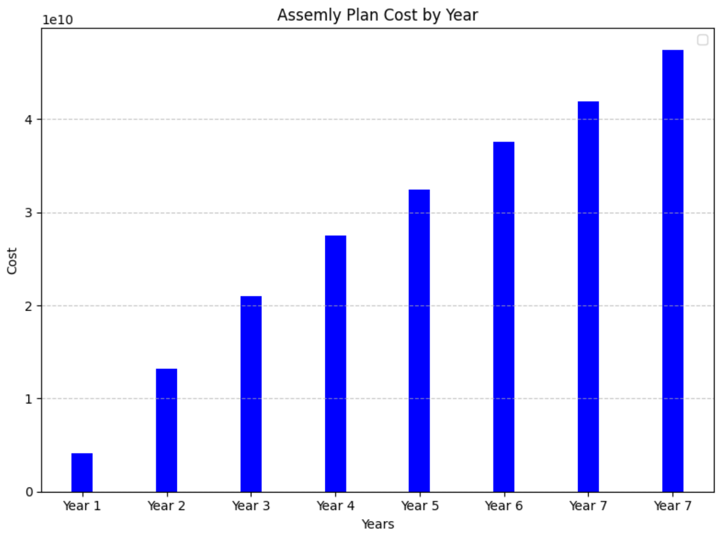


Figure 31: Assembly Line Costs by Year

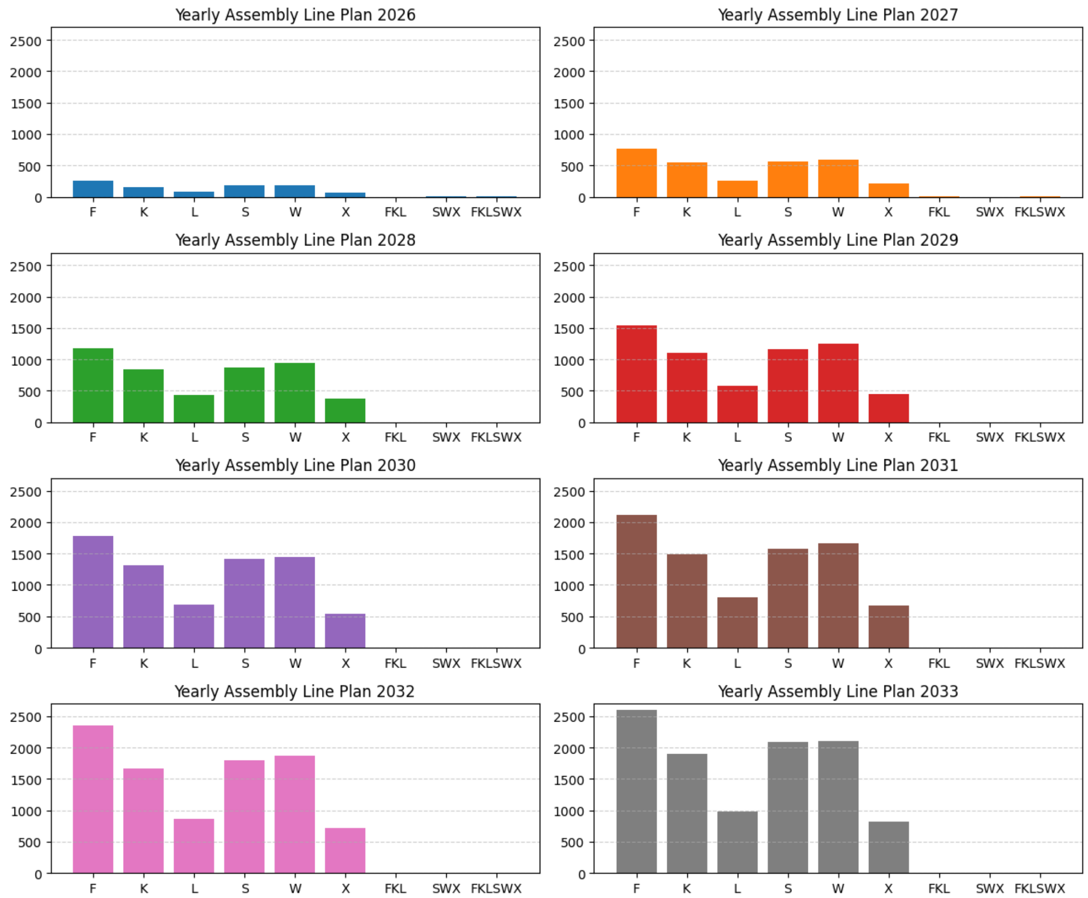


Figure 32: Assembly Line Plans by Year

2.6 Estimation of the supply warehouse's required capacity

We first collect data on the number of pallets required at the warehouse each year derived from data from Task 1.4 and Task 2.1. We also collect data on the maximum throughput of trucks and pallets at the warehouse each year to estimate the number of dock doors and the number of workers required each year.

2.6.1 Storage Capacity Estimation

Based on data from Task 1.4 and Task 2.1, we calculate the required storage capacity for each product at the warehouse each year. Combining the given data of product boxing sizes and two-high pallet sizes, we calculate the maximum number of pallets required at the warehouse each year to store maximums of two-week supplies from Europe and four-week supplies from China. In doing so, we only use wooden pallets that are type-indifferent. The results are shown in the attached file *:/Task 2/Task 2.6/2.6 Storage Capacity.csv*

2.6.2 Throughput Estimation

We calculate the throughput in mixed-product pallets and trucks, including incoming products from European and Chinese suppliers and outgoing supplies to the assembly center.

In estimating the number of mixed-product pallets, we consider a combination of vacancy and volumetric analysis (total volume of pallets vs. total volume of the products) with a vacancy factor of 1.2. This means the required volume of pallets is 1.2 times the total volume of products, and the vacancy factor efficiently captures the vacant spaces that would appear when loading pallets.

We used the pallet throughput estimation to estimate the truck throughput. We assume that the warehouse/assembly center-to-DC trips use **semi-trailer trucks** with capacities of 30 pallets, and suppliers-to-warehouse trips also use **semi-trailer trucks** with capacities of 30 pallets. These capacities are based on the online resources we gathered. Based on the given guideline, outgoing trucks are FC-specific.

The results are shown in the attached file *:/Task 2/Task 2.6/2.6 Throughput.csv*

2.6.3 Warehouse Rack Assignment

We calculate the number of 8-height racks for the warehouse by years shown in Figure 33. As the figure shows, the number of racks required at the warehouse increases as the year increases.

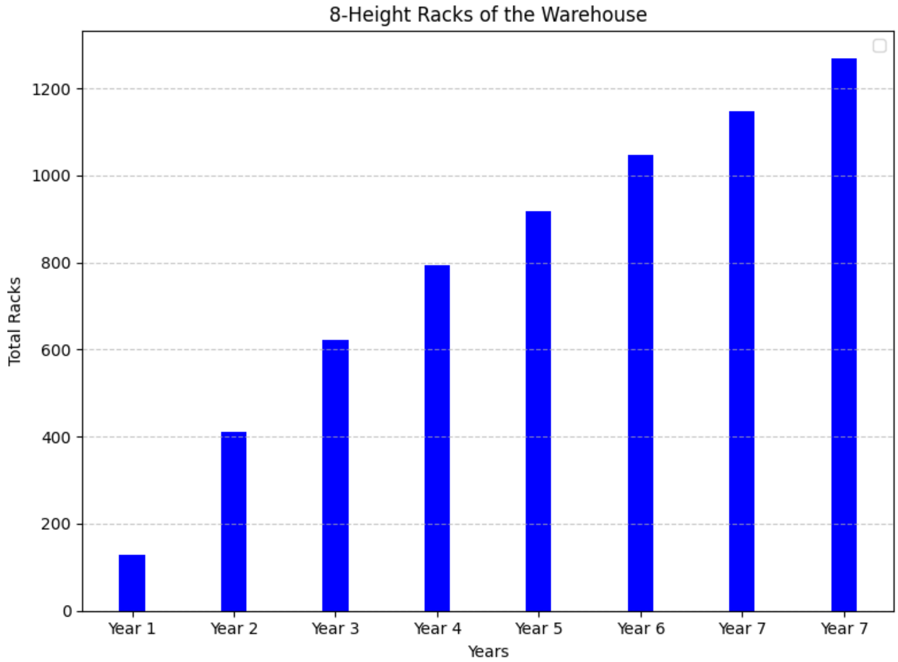


Figure 33: 8-Height Racks of the Warehouse by Years

2.6.4 Warehouse Dock Door Assignment

In estimating the number of dock doors of the warehouse each year, we combine the truck throughput data and the tier plan data, and we assume:

- The warehouse operates 12 hours a day.
- Incoming and outgoing trucks share the same set of dock doors.
- It takes one hour to load and unload a truck. Thus, a dock door can handle 12 trucks per day at best.

These assumptions lead us to calculate the number of dock doors required at the warehouse each year in order to accommodate the truck throughputs. The results are shown in the attached file *:/Task 2/Task 2.6/Warehouse Plans.csv*. Figure 34 visualizes the result. We want to point out that sometimes, the DC requires fewer dock doors as the year increases. This is because our generated simulation data on daily demand has some variations and is not strictly mono-increasing by year.

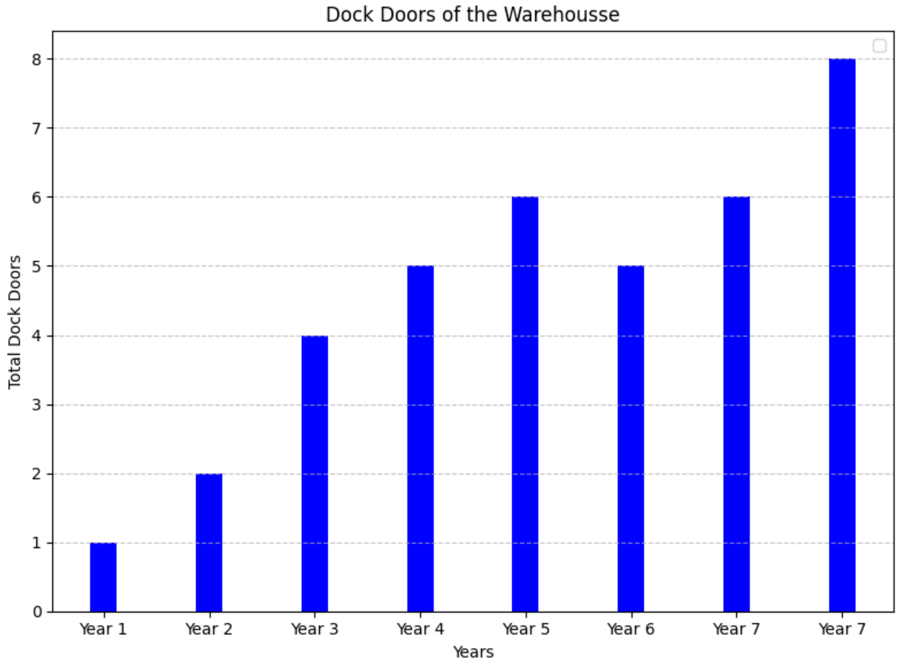


Figure 34: Dock Doors of the Warehouse by Years

2.6.5 Warehouse Worker Assignment

In estimating the number of workers of the warehouse each year, we combine the pallet throughput data and the tier plan data, and we assume:

- The warehouse operates 12 hours a day.
- Each rack takes $1.5m \times 1.5m$ spaces, which is large than the pallet size of $1.2m \times 1.2m$ accounting for walking and loading spaces.
- The overall size of the warehouse is $1.5\times$ the size of its racks, accounting for docking, resting, and working spaces.
- The docks surround the racks in the shape of a square. Thus, the average walking distance between dock doors to the racks is half of the side of the DC, and since the worker will need to travel back, the average walking distance for managing a pallet is the side of the warehouse.
- A worker can only handle one pallet at a time, and it takes 15 minutes for each worker to locate/load/organize a pallet.
- Together, the average time it takes for a worker to finish all processes associated with a pallet is $\sqrt{\lceil (max\_pallets/8) \rceil * 1.5 * 1.5 * 1.5}/82 + 15$ minutes.

- So a worker can handle is $\lfloor 12 * 60/(\sqrt{\lceil(max\_pallets/8)\rceil * 1.5 * 1.5 * 1.5}/82 + 15)\rfloor$ pallets per day on average

These assumptions lead us to calculate the number of workers required in the warehouse each year to accommodate pallet throughput.re shown in the attached file *:/Task 2/Task 2.6/Warehouse Plans.csv*. Figure 35 visualizes the result. We want to point out that sometimes, the warehouse requires fewer workers as the year increases. This is because our generated simulation data on daily demand have some variations and are not strictly mono-increasing by year.

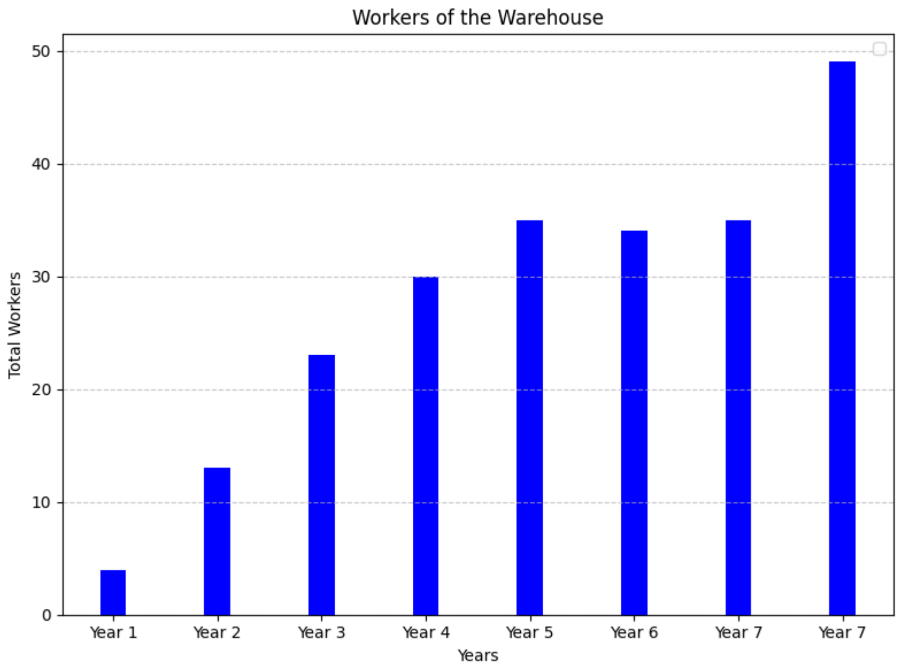


Figure 35: Workers of the Warehouse by Years

2.7 Estimation and picture of BotWorld's inter-facility flows

The computations for this part have been done in the order : DC to FC then Plant to DC then Supply Warehouse to Plant. We have started with the more complicated flows so as to be able to reuse functions from the python code in the more simple ones. This explains the current organization of this task, which might not seem logical at first.

2.7.1 DC to FCs flows

In this part we will focus on the flow from the DC to the FC. First, we will start by computing the yearly demand each of these FC should be able to hold. We have coded this in *:/Task 2/Task 2.7/task2_7_YearlyDemand.py*. The output of this code is an excel file *:/Task 2/Task 2.7/bot_demand_analysis.xlsx* showing the yearly demand in each of the markets for each product in the sheets *Cities_i* with i in {2026, ..., 2033} and each FC in the sheets *FC_i* with i in {2026, ..., 2033}.

We will use this first output to determine the flow in terms of number of MyBots in the code *:/Task 2/Task 2.7/task2_7_Main_DC_FC.py*. This output is plotted in figure 36.

In this code, we also compute the **volume** and **weight** this represents using the table 8. The output is also plotted in the same graph in figure 36.

CityBot		Shipping box dimensions			
Category	Model	Length (cm)	Width (cm)	Height (cm)	Weight (kg)
Floor Care	F10	55	55	50	19
	F20	55	55	50	19
	F30	65	55	50	25
	F50	70	60	50	32
Kitchen Help	K10	50	40	50	12
	K20	50	40	50	12
	K30	55	45	50	15
	K50	60	45	50	18
Leisure	L20	60	40	50	18
	L50	60	50	50	20
Safety & Security	S10	70	40	55	22
	S20	70	40	55	22
	S30	70	45	55	25
	S50	70	55	55	29
Wall & Window Care	W10	45	45	45	12
	W20	45	45	45	12
	W30	55	45	45	14
	W50	55	55	45	16
Exterior Care	X20	65	70	60	43
	X50	75	80	65	60

Table 8: CityBot shipping box dimensions

Finally, we will focus on flows in terms of **pallets**. To do so, we will consider the 2 different sizes of pallets we can use, assuming we will use the same pallet for all models, in order to simplify the computations. That means, if we were to chose a Wooden pallet – 3 runner for Model F10, then all F10 models will be sent on this kind of pallet for the whole year. Though, for all models, we will chose the type of pallet that is the most space efficient, considering we can only stack two levels of models on the same pallet. We have also added height and weight criterias on the pallet, so that the pallet cannot carry more than a 1000kg or be higher than 2 meters. Assuming all this, we have computed the number of pallets we will need to ship from the DC to each FC every year.

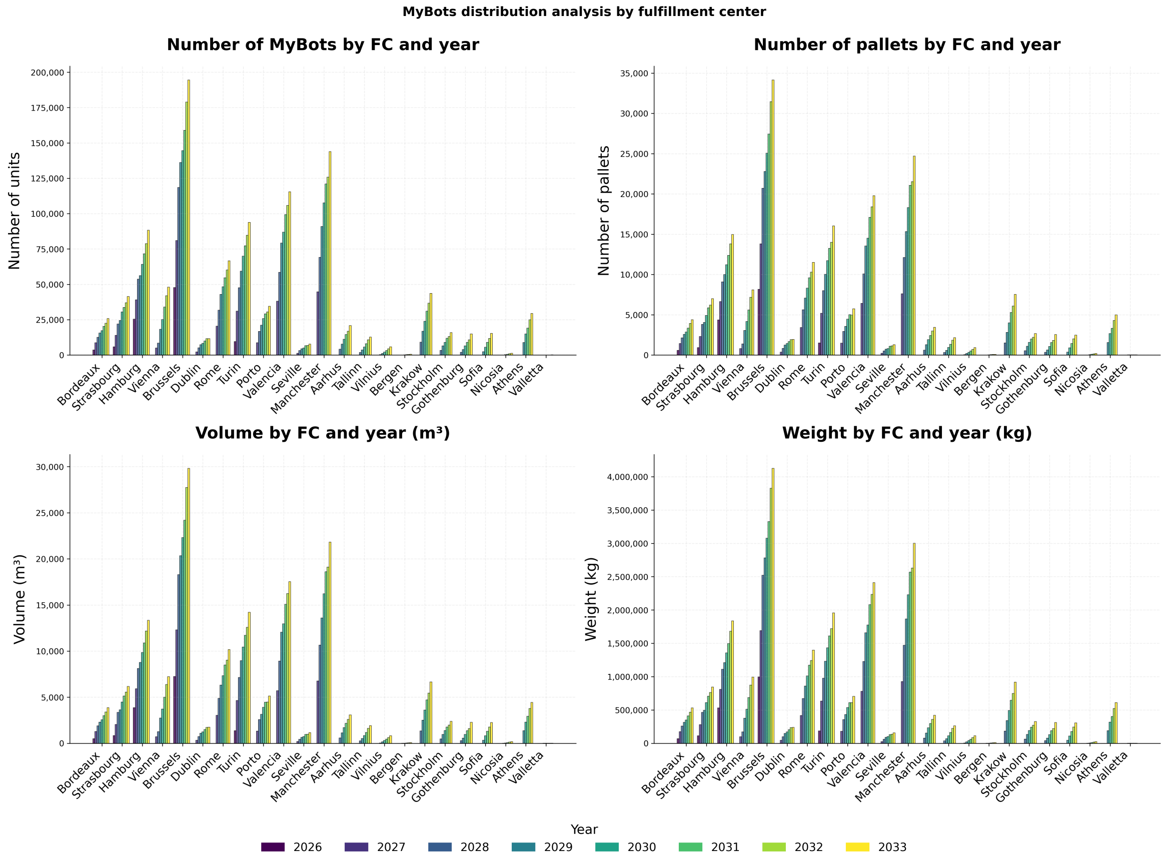


Figure 36: DC to FCs flows from 2026 to 2033

The output, displayed in figure 36 enhances that several FCs such as Valletta, Bergen, Nicosia help fulfill a negligeable demand compared to bigger ones such as Brussels. This might question, whether it is financially worth it to have them in the Network. Do we really need to penetrate the Cypriot network and operate a FC there for such a low demand? Should we have a FC in Oslo only for the whole Northern region, even if that means begin more than 5 hours away to all markets?

2.7.2 Plant to DC flows

The code used for this part, *:/Task 2/Task 2.7/task2_7_Main_Plant_DC.py*, uses some function of the next one to define the number of pallets, weight and volume. The output is displayed in figure 37.

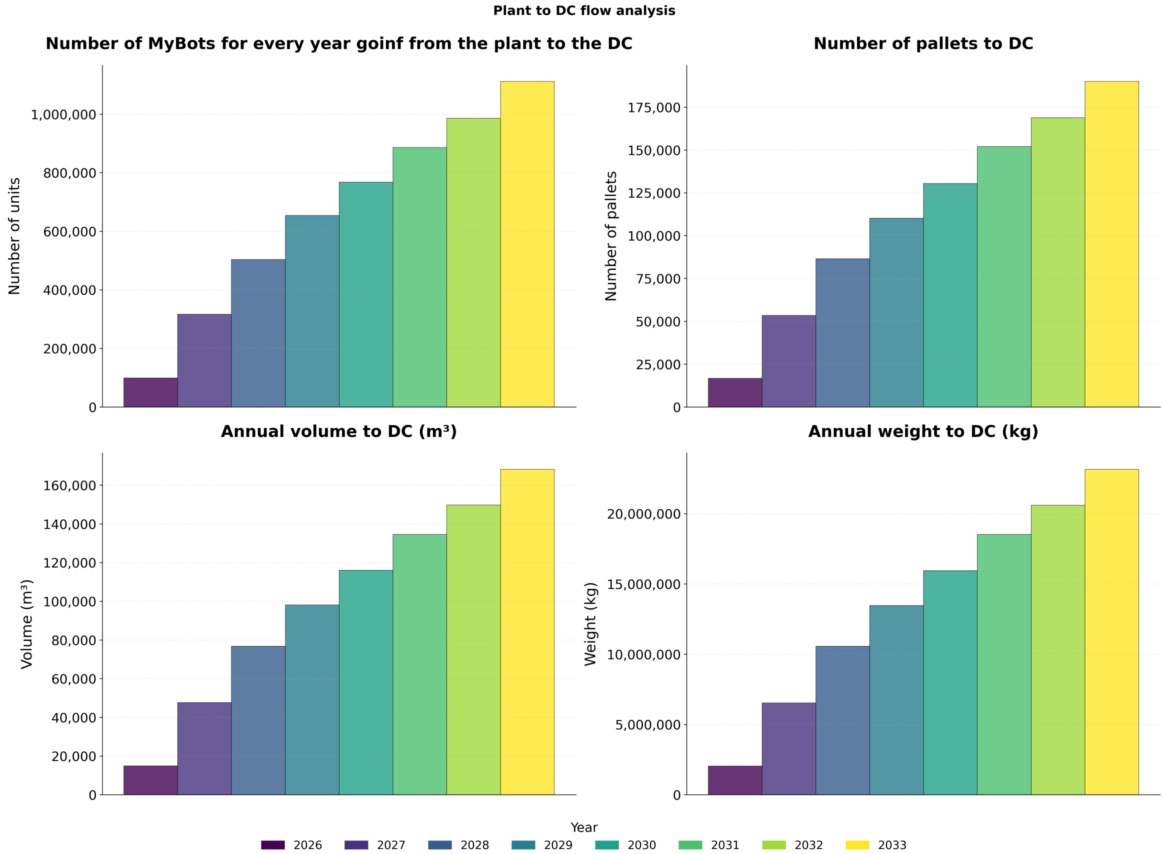


Figure 37: Plant to DC flows from 2026 to 2033

We can make several observations on the graphs:

- Consistent growth pattern:
  - Each metric (units, pallets, volume, weight) shows a steady year-over-year increase.
  - From 2026 to 2033, the total production volume increases by approximately 8-9 times.
- Scale of operations:
  - By 2033, the plant will need to produce over 1 million units annually

  - The pallet movement will reach around 175,000 pallets per year
  - The annual volume will exceed 160,000 m^3
  - The weight handled will be over 20 million kg (20,000 tonnes)

- Operational implications:
  - The steep increase in all metrics suggests a need for scalable infrastructure
  - The DC will need significant expansion capacity to handle the 2033 volumes
  - Material handling systems must be designed for the peak 2033 capacity, which is about 9 times the initial 2026 requirements

- Logistical challenges:
  - The growing number of pallets (from 20,000 to 175,000 annually) will require sophisticated pallet management
  - The weight increase to 20 million kg will impact transportation requirements and loading dock specifications
  - Volume growth to 160,000 m^3 has implications for storage space and warehouse design

2.7.3 Supply Warehouse to Plant flows

For this part, the main change is that *"From the supply side, production requires parts and materials from multiple suppliers. In average, these are twice denser than the final product."* So we will use the following table (table 9) for the volume and weight computations considering that the weight is the same but that the volume is twice smaller than that of the packaged product.

CityBot		Packaged Dimensions			
Category	Model	Length (cm)	Width (cm)	Height (cm)	Weight (kg)
Floor Care	F10	40	40	30	15
	F20	40	40	30	15
	F30	48	40	30	20
	F50	53	44	30	25
Kitchen Help	K10	32	25	28	8
	K20	32	25	28	8
	K30	36	28	28	10
	K50	42	30	28	12
Leisure	L20	45	23	30	12
	L50	45	33	30	16
Safety & Security	S10	56	26	33	18
	S20	56	26	33	18
	S30	56	32	33	20
	S50	56	38	33	23
Wall & Window Care	W10	30	30	25	8
	W20	30	30	25	8
	W30	40	30	25	9
	W50	40	40	25	11
Exterior Care	X20	50	55	40	35
	X50	60	65	45	50

Table 9: CityBot packaged dimensions

We use the following code for this part: *:/Task 2/Task 2.7/task2_7_Main_Supply_Plant.py*. The result is plotted in figure 38.

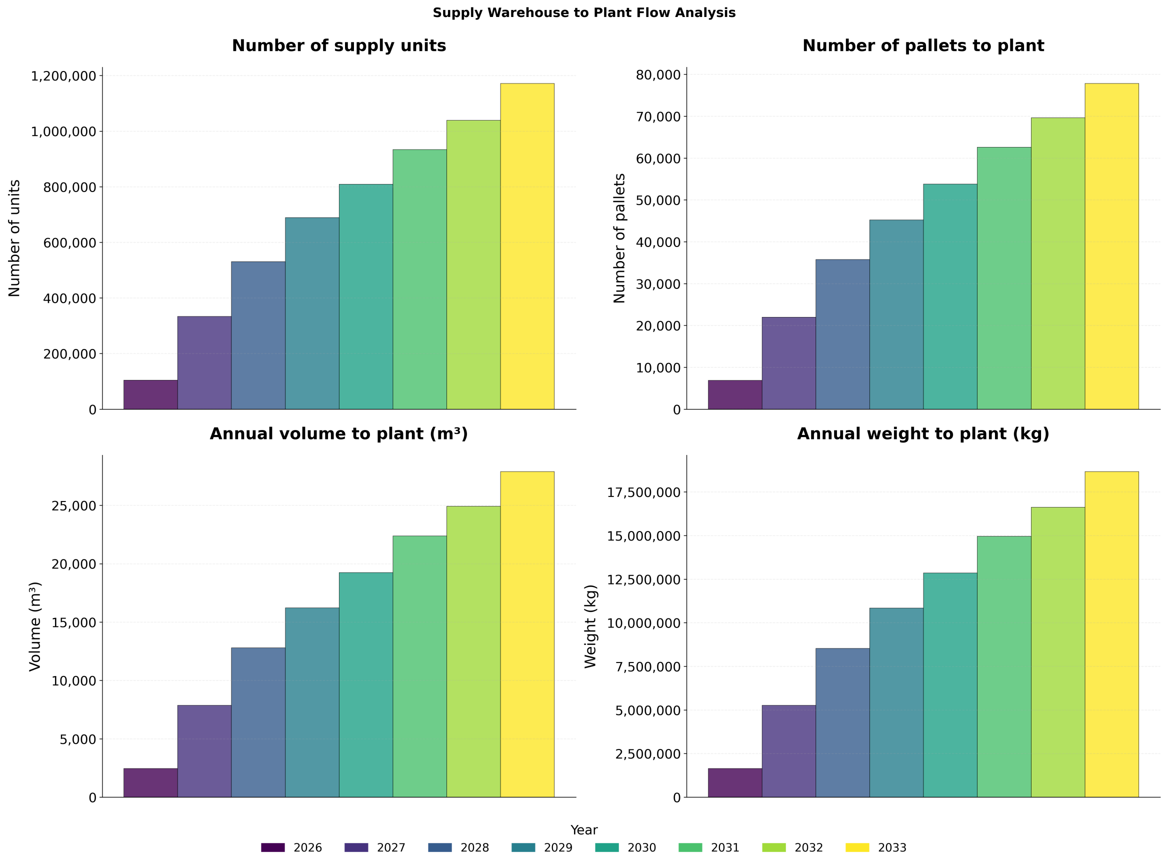


Figure 38: Supply Warehouse to Plant flows from 2026 to 2033

While unit count reaches 1.2 million in 2033 (similar to finished products), total volume is only 27,000 m^3 thanks to doubled component density, enabling optimization of storage and transport space. Compared to plant-DC flow (175,000 pallets), supply warehouse-plant flow only requires 77,000 pallets in 2033, demonstrating the efficiency of component packaging.

2.8 Fleet proposal in each year of the 2026-2033 horizon

In this part we have made as before the following assumptions:

- **Semi trucks** will circulate between the Suppliers and the Supply Warehouse. Semi-trucks can hold up to **30 pallets**.
- The Supply Warehouse, the Plant and the DC are located in the same place, so we will not need any trucks to move things around.
- **Heavy rigid truck** will circulate between the DC and the FCs. Heavy rigid trucks can hold up to **20 pallets**.
- **Light rigid truck** will circulate between the FCs and the markets they serve (see table 6). Light rigid trucks can hold up to **14 pallets**.

We will compute gas emission and gas consumption using the data found on the website of www.transportpolicy.net

Truck	Emission (g/km)	Consumption (l/100km)
Semi trucks	1100	45
Heavy rigid truck	900	35
Light rigid truck	450	17.5

Table 10: Gas emission and consumption for the trucks we consider

Regarding the trucks and drivers, we will consider:

- Drivers can drive up to **9 hours a day**.
- The trucks will always come back empty after being unloaded. We will need to account for that travel time too.
- If the travel time exceed 9 hours a day, we will add an additional trucker, above 18 hours a day, a 3rd trucker will be added too.
- Drivers will drive at an average speed of **80 km/h**.
- We will not consolidate trucks, we will fill them every week/day with what need to be sent and compute the number of trucks based on this assumption.
- Once the truck and its trucker(s) are back to their initial location, they are available to be reemployed.

2.8.1 Count of Semi-Trucks delivering supplies to the Warehouse

Number of pallets coming from each supplier every week

It is stated the following: *The Chinese supply is to be consolidated in Shenzhen and brought in containers by ships arriving at the Port of Rotterdam. The European suppliers are based in the Haut-de-France in France, the Upper Austria region of Austria, the Pomerania region of Poland and the Andalusia region of Spain, with basically a fourth of the flow from each country.* So we will assume that the supplies come from the following cities (11, as we already have them in our driving distance matrix established for 2.1.1.

City	% of supplies
Rotterdam	40%
Linz	15%
Lille	15%
Warsaw	15%
Seville	15%

Table 11: China and EU suppliers and their percentage of supplies

In the csv file *:/Task 2/Task 2.8/Warehouse Weekly Throughput From CN and EU Supply.csv*. We have the total throughput from China and EU supply at the warehouse. We will compute the number of pallets this represent from each of the suppliers so as to determine how many trucks will be required. This is coded in `:/Task 2/Task 2.8/task2_8_suppliers_to_warehouse_pallets.py`.

Semi-truck fleet size analysis

Knowing how many pallets come from each of these 5 cities every week, we will be able to send as many trucks and drivers, according to the assumptions made here-before 2.8. This is coded in `:/Task 2/Task 2.8/task2_8_suppliers_to_warehouse_semi.py`. The output is displayed in *:/Task 2/Task 2.8/Semi_fleet_requirements.xlsx*.

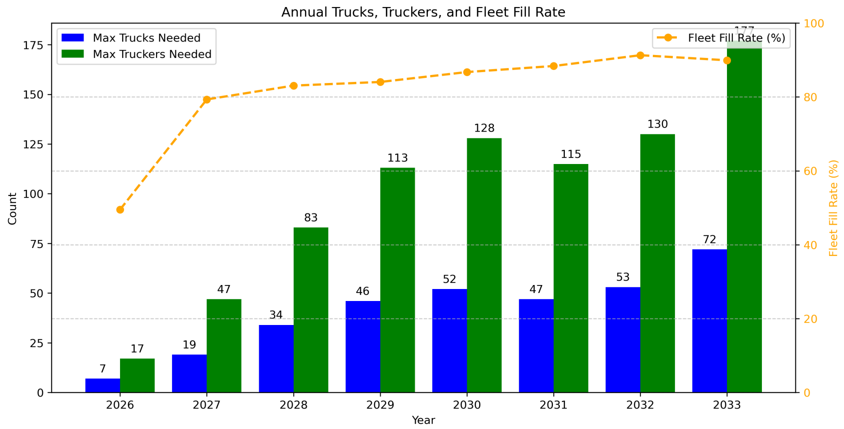


Figure 39: Semi trucks fleet size

This graph effectively visualizes the estimated number of semi-trucks required for supplier-to-warehouse transportation. It accounts for varying supply volumes from different European and Chinese suppliers over the 2026-2033 horizon. We observe that the number of trucks needed doesn't necessarily grow as the years pass, which was not expected. This is probably due to the method we use to compute the number of pallets. Indeed we consider the number of pallets that should arrive every week at the warehouse then we attribute each one of these pallets to a given suppliers considering the table 11, and we round up this number, which may result in an over estimated number of pallets. Plus we do not consolidate shipments so there are weeks when we can have a truck that is filled with only 1 pallet.

Overall, we notice that we need a lot more drivers than trucks, meaning there are trucks that have 3 truckers inside. This is due to the assumption that we add one driver to the truck if we reach the time limit a driver can drive a day. This could be improved by having maybe more trucks and less drivers if workforce is expensive.

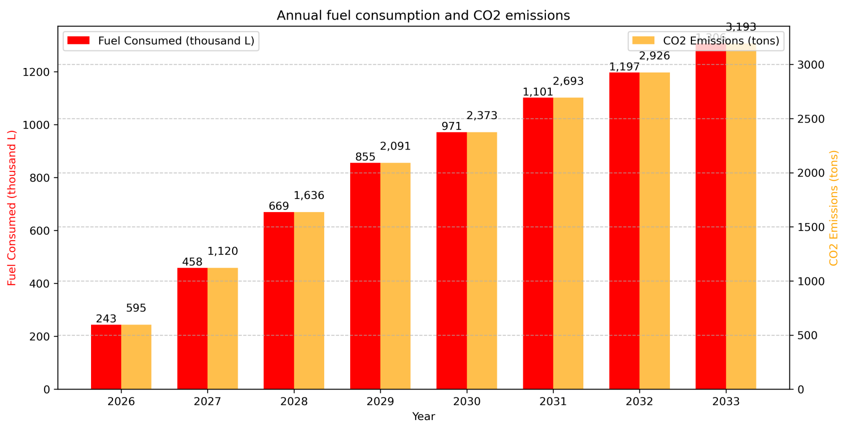


Figure 40: Environmental impact of semi trucks

This graph accurately quantifies the CO2 emissions and fuel consumption of semi-trucks. The emissions calculation is based on standard values for diesel-powered freight transport (1,100 g/km CO2, 45 L/100km fuel consumption). We note, without any surprise, that the fuel consumption and gas emissions linearly raise through the year since we increase the number of trips and trucks.

2.8.2 Count of Heavy rigid trucks delivering Packed MyBots from the DC to the FCs

Number of pallets going to the FCs every week

FCs are to receive weekly deliveries of packed MyBots from the DC. We will start by counting how many pallets that represents every week. To do so we will use the output of Task 2.4, (*task2.4_output_weekly.xlsx*) that computed a weekly demand for MyBots. We will consider 2 weeks of robust inventory so we will start by ordering a lot of MyBots the first week to start creating the inventory then we will receive deliveries 2 weeks in advance of the forecasted demand (obtained by the output of 2.4).

We will not consolidate the pallet at this stage and will consider that one pallet is dedicated to one Model of MyBot, and that we will stack up to 2 Bots. We will try to get the most optimal size of pallet every time so as to fill the pallet as much as possible, choosing between Wooden pallet – 3 runner and the Perimeter wooden pallet.

This has been coded in `:/Task 2/Task 2.8/task2_8_DC_to_FC_pallets.py` The output is displayed in *:/Task 2/Task 2.8/pallets_per_week.xlsx*. In this output we know how many pallets have to be moved from the DC to the FCs every week. We will use this as an input for the next part.

Heavy fleet size and analysis

We will sent several drivers and several trucks every week from the DC to the FCs. We will count the number of trucks we need to send every day and see when they come back. Since we send trucks every week, we will most likely be able to send the same truck the next week if necessary, because it would have had time to do the roundtrip. Nevertheless, if we don't have enough trucks back at the beginning of the following week, we will need to add more trucks and truckers to our fleet. This is what the code does. The number of drivers is computed based on the assumption that we add one more driver every 9 hours (maximum daily limit of driving in Europe), so one truck can have up to 3 drivers.

This has been coded in `:/Task 2/Task 2.8/task2_8_DC_to_FC_heavy.py`. The output is displayed in the file: *:/Task 2/Task 2.8/Heavy_fleet_requirements.xlsx*

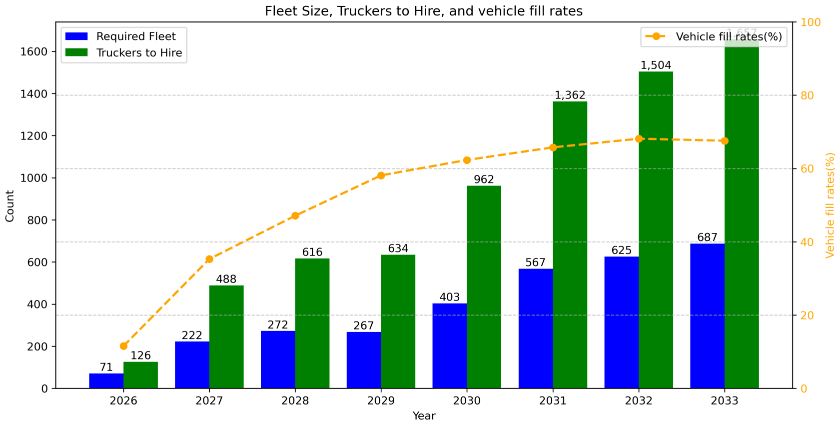


Figure 41: Heavy trucks fleet size

The fleet size starts at 71 trucks in 2026 and rises to 687 by 2033. This increase is expected because DC-to-FC shipments occur once per week, and the number of FCs increases over time to maintain delivery nearness. The assumption that each truck carries 20 pallets and that FCs stock two weeks of inventory results in a feasible fleet size. The smooth increase over the years follows the MyBot demand curve, ensuring that FCs receive just-in-time replenishment without excess capacity.

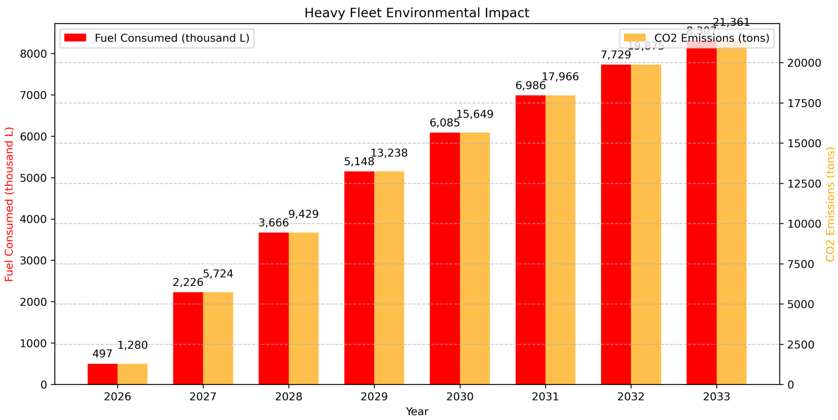


Figure 42: Environmental impact of heavy trucks

CO2 emissions and fuel consumption follow a gradual upward trend, with initial emissions at around 1,300 tons of CO2 in 2026, reaching 21,000 tons by 2033. This aligns well with the assumptions on mileage per truck and average truck utilization rates. The numbers validate the fleet sizing approach and confirm that our logistics design efficiently distributes inventory while maintaining a balanced environmental impact.

2.8.3 Count of Light rigid trucks delivering Packed MyBots from the FCs to the markets

Number of pallets going to the markets every day

In this part we will use the output for task 1.4, to determine the number of pallets that need to be shipped every day to each market. In this part we will consolidate pallets, this means we can put different models on the same pallet, so as to minimize the number of pallet we need to ship every day.

This has been coded in `:/Task 2/Task 2.8/task2_8_FC_to_market_light_demand_daily.py`. The output is displayed in *:/Task 2/Task 2.8/daily_pallet_requirements_by_market.xlsx.*

Light fleet size and analysis

As in subsubsection 2.8.2, we will compute the number of light trucks we need in our fleet. Knowing the number of pallets each market should receive every day and which FC should supply which market we will send as many trucks and truckers as needed every day and reuse them as soon as they are back to the FC.

This has been coded in the file `:/Task 2/Task 2.8/task2_8_FC_to_market.py`. The output is displayed in the file *:/Task 2/Task 2.8/Light_fleet_requirements.xlsx*

The results are displayed here after.

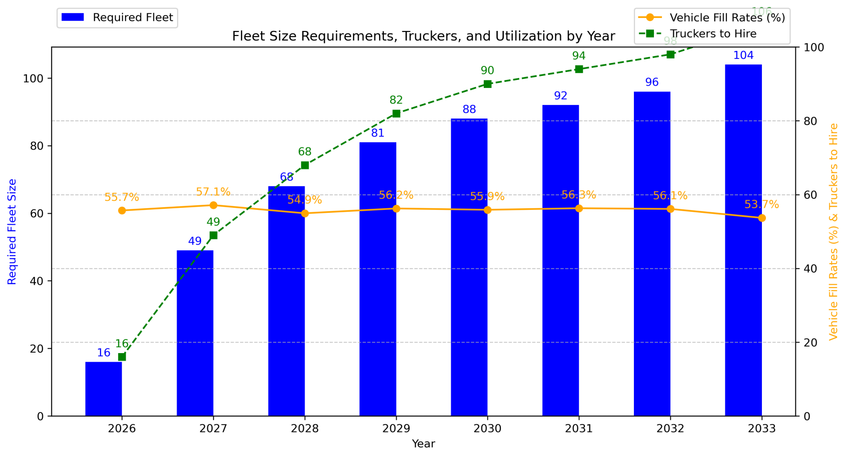


Figure 43: Light trucks fleet size

The fleet starts with 16 light trucks in 2026 and expands to 104 by 2033. This increase makes sense given that light trucks handle last-mile deliveries, where demand is highest. The assumption that each truck carries 14 pallets and that they perform multiple delivery rounds per day is reflected in these numbers. The steady fleet growth is proportional to demand expansion in new metropolitan areas, ensuring that next-day delivery service levels are met across Europe.

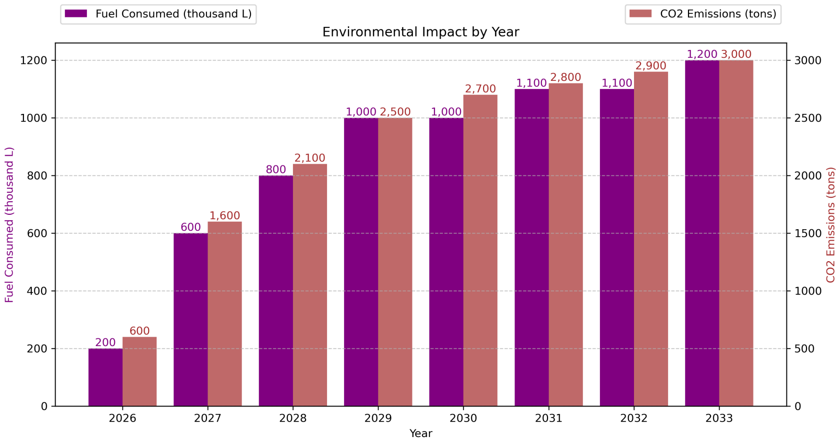


Figure 44: Environmental impact of light trucks

CO2 emissions begin at 600 tons in 2026 and rise to 3,000 tons by 2033, reflecting the increasing number of urban deliveries. Since these vehicles operate in stop-and-go traffic conditions, their fuel consumption per km is slightly higher, reinforcing the assumption that city distribution requires a higher fleet density compared to long-haul transport. The numbers confirm that fleet sizing is correctly balanced to ensure efficient customer deliveries.

2.8.4 Complete fleet analysis

In this part we will aggregate all previous data to get more insights. The computations have been done in `:/Task 2/Task 2.8/task2_8_total.py` and `:/Task 2/Task 2.8/task2_8_total_CO2.py`.

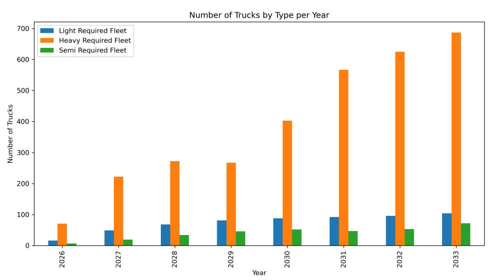


Figure 45: Total number of each type of trucks per year

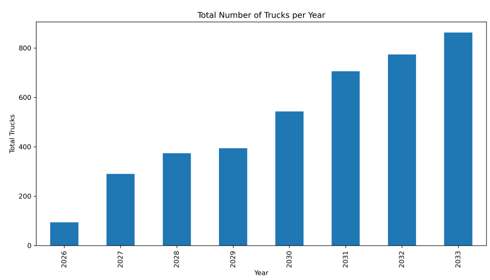


Figure 46: Total fleet size per year

Figure 45 shows the total number of each type of truck per year. Between 2026 and

2033, semi-trucks remain relatively stable due to consistent inbound supply needs, while the number of heavy trucks transporting goods from the distribution center (DC) to fulfillment centers (FCs) rises from 71 to 687. This increase follows the expansion of the FC network and the growing demand for MyBots.

Figure 46 aggregates the total fleet size per year, showing a smooth increase in the number of trucks from approximately 300 in 2026 to over 1,300 by 2033. This aligns well with projected MyBot demand, ensuring that capacity scales appropriately without excessive investment in unused vehicles. The fleet growth is essential for maintaining just-in-time replenishment strategies while keeping distribution efficient.

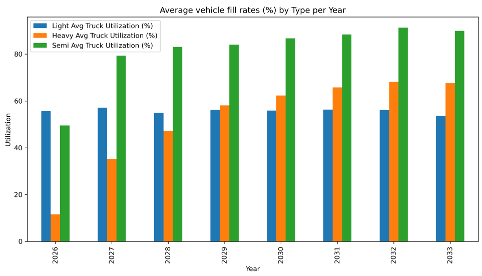


Figure 47: Average vehicle fill rate per type of truck per year

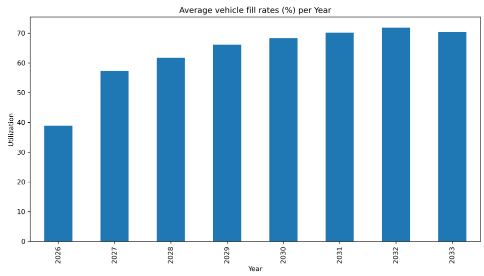


Figure 48: Average vehicle fill rate per year

Figures 47 and 48 examine the average vehicle fill rate per truck type and overall fleet efficiency. The fill rate trends suggest that heavy trucks and light trucks are generally well-utilized, but semi-trucks show some inefficiencies. This is likely due to the method of pallet calculation, which may be slightly overestimating weekly needs, leading to trucks sometimes departing with minimal loads. Optimizing consolidation and improving load planning could increase fill rates and reduce unnecessary trips. In the first year, the average vehicle fill rate is very low. This is due to the fact that just for one pallet, we might add a second truck on the road to a specific FC. Maybe we could have considered light trucks for this transfer so as to fill them more.

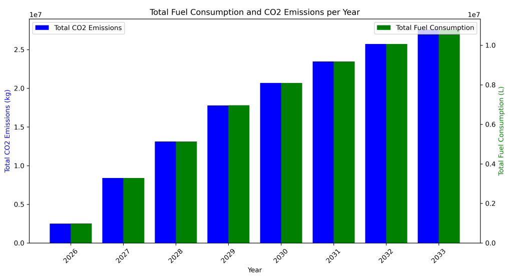


Figure 49: Total fuel consumption and greenhouse gas emission per year

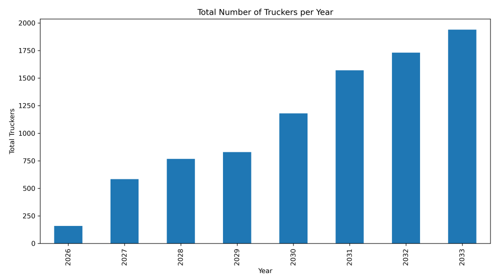


Figure 50: Total number of truckers per year

Figure 49 presents total fuel consumption and greenhouse gas emissions per year. The numbers rise proportionally to fleet size, with total CO2 emissions starting at around 3,000 tons in 2026 and reaching nearly 25,000 tons by 2033. This increase is expected given the reliance on diesel-powered vehicles and the growing demand for deliveries. However, it highlights a crucial issue: without alternative fuel strategies or increased efficiency in routing, environmental impact will continue to escalate. This is where we might consider using different vehicles, electric or hydrogen driven for instance. Amazon has actually acquired a lot of Rivian electric trucks (Everything you need to know about Amazon's electric delivery vans from Rivian), showing some hope for the future!

Figure 50 shows the total number of truckers required per year. The number of truckers increases significantly, surpassing fleet growth due to the assumption that additional drivers are needed when trip durations exceed daily driving limits. If workforce is expensive, we might want to reduce the number of drivers, even if that means increasing the number of trucks. Indeed in this case, we allowed 3 drivers to be on the same truck to keep the truck moving, which might be a huge waste of money.

2.9 Expansion of the simulator to model daily aggregate operations of the planned capacitated factory, supply warehouse, distribution center, and fulfillment centers operations, and fleet vehicles

The objective of this task was to expand the existing simulator to model the daily aggregate operations of a planned capacitated factory, a supply warehouse, a distribution center (DC), fulfillment centers (FCs), and fleet vehicles. The expansion aimed to integrate a detailed trucking schedule and fleet management system to optimize logistics and resource allocation. The detailed schedules are available in the Task 2.9 file.

2.9.1 Data Preparation

The input dataset, sourced from daily pallet requirements across various markets (*Task2.8/daily_palet_req* was processed to extract relevant information (daily pallet demand). Each fulfillment center (FC) serves specific markets, and a data cleaning function was implemented to restructure the dataset for analysis.

2.9.2 Trucking Schedule Implementation

Two scheduling models were developed:

- **Truckload (TL) Model:** This model consolidates shipments until a full truckload (14 pallets) is reached before dispatch.

```
Date: 2026-02-08
Paris accumulated pallets: 14
Departures for Paris: 1
Brussels accumulated pallets: 0
Departures for Brussels: 0
Lille accumulated pallets: 14
Departures for Lille: 1

Date: 2026-02-09
Paris accumulated pallets: 10
Departures for Paris: 0
Brussels accumulated pallets: 0
Departures for Brussels: 0
Lille accumulated pallets: 1
Departures for Lille: 0

Date: 2026-02-10
Paris accumulated pallets: 16
Departures for Paris: 1
Brussels accumulated pallets: 0
Departures for Brussels: 0
Lille accumulated pallets: 2
Departures for Lille: 0
```

Figure 51: Truckload Model: consolidation and shipment with a few cities

- **Less-than-Truckload (LTL) Model:** In this model, at least one truck is dispatched daily if there are outstanding pallets, even if the truck is not full. This model is simpler, since we do not have to take into consideration the accumulation of pallets. However, it may raise concerns regarding its actual feasibility: do we need to pay for transportation cost to deliver only one pallet?

```
Date: 2026-09-20
Departures for Toulouse: 1
Departures for Nantes: 1
Departures for Bordeaux: 1
Departures for Strasbourg: 0
Departures for Munich: 1
Departures for Frankfurt: 1
Departures for Stuttgart: 1
Departures for Zurich: 0
Departures for Berlin: 1
Departures for Hamburg: 1
Departures for Vienna: 0
Departures for Graz: 0
Departures for Linz: 0
Departures for Zagreb: 0
Departures for Prague: 0
Departures for Budapest: 0
Departures for Bratislava: 0
Departures for Ljubljana: 0
Departures for Brussels: 0
Departures for Paris: 1
Departures for Lille: 1
Departures for Cologne: 1
Departures for Antwerp: 0
Departures for Ghent: 0
```

Figure 52: Less than Truckload Model: no consolidation

For both models, the number of trucks departing per FC per day was calculated, and cumulative pallet accumulation was tracked.
We will not expand further the TL Model and will focus on the LTL Model, as this has been the primary assumption for this casework.

2.9.3 Fleet Sizing and Management

To estimate the required fleet size, a function was implemented to track active trucks and their return times. The maximum number of active trucks over the simulation period determined the necessary fleet size. The methodology was applied separately for FCs and the distribution center (DC), incorporating specific transit days for each FC. We suggested that to offer same day delivery, the light trucks had a transit day of 0, which is not the case for the medium trucks (depending on the distance from Metz)
In order to calculate the number of transit days (or cycle time) for a medium truck from

DC to FC x:

$$\Delta_x = \lceil \frac{D(DC, x)}{c_M} \times 2 \rceil$$

With: Δ_x: the transit time for a medium truck from DC to x (in days)
$D(DC, x)$: the total distance from DC to x
c_M: the speed of the truck (80 km/h)
The factor 2 doubles the distance because we have to take into account the return time for the driver(s) and the truck.

2.9.4 Assumptions

Several assumptions were made to streamline the simulation, similarly as in Task 2.8:

- Each truck can carry a maximum of 14 pallets (for FC shipments), 20 pallets (for DC shipments), and 30 pallets (for supplier shipment).
- LTL shipments guarantee at least one daily departure, ensuring demand is met in a timely manner.
- Transit times vary from DC to FC, ranging from 0 to 7 days.
- Fleet size calculations assume immediate truck availability upon return.
- The distribution center receives and redistributes pallets without delay.

The simulation generated daily truck departures for each FC and DC, which were exported for further analysis. Additionally, the estimated fleet size required for FC and DC.

We kept `print()` commands under comments so that the reader can see into detail how does the code executes itself.

Date	Munich	Berlin	Paris	Lille	Cologne	...
01/01/2026	0	1	1	0	0	...
02/01/2026	0	1	1	0	0	...
03/01/2026	1	1	1	0	1	...
04/01/2026	1	1	1	1	1	...
05/01/2026	1	1	1	1	1	...
06/01/2026	1	1	1	0	1	...
07/01/2026	0	1	1	0	0	...
08/01/2026	0	1	1	0	0	...

Table 12: Daily Departures to Various Cities

Date	Bordeaux	Strasbourg	Hamburg	Vienna	Brussels	...
01/01/2026	0	0	1	0	1	...
02/01/2026	0	0	1	0	1	...
03/01/2026	0	1	1	0	1	...
04/01/2026	1	1	1	0	1	...
05/01/2026	1	1	1	0	1	...
06/01/2026	0	1	1	0	1	...
07/01/2026	0	0	1	0	1	...
08/01/2026	0	0	1	0	1	...
09/01/2026	0	0	1	0	1	...

Table 13: Departures from DC to FCs

For the heavy trucking schedule, we did slightly differently. Since we are retrieving goods from our supplier, we distributed equally the weekly pallet transportation schedule (also retrievable from task 2.8) and distributed the demand uniformly across the days of the week.

Date	Trucks
01/01/2026	1
02/01/2026	1
03/01/2026	0
04/01/2026	0
05/01/2026	0
06/01/2026	0
07/01/2026	0
08/01/2026	1
09/01/2026	1

Table 14: Supplier to Warehouse

If we wanted to implement a *just in time* strategy, we could have shifted the schedule up for the heavy and medium trucks by $\frac{1}{2}$ the transit time. However, since we considered a safety stock that is capable of absorbing the demand, we will not explore this concept any further.

2.9.5 Determining the fleet size from the daily truck schedule

Expanding from the daily truck schedule Using the data calculated in the previous subsections of Task 2.9, we can find the number of small, medium and heavy ships necessary for the entire supply chain to work accordingly. The functions `calculate_fleet_size` and `calculate_DC_fleet_size` are implemented to do the exact same thing: for each day, take the number of trucks that need to depart as well as the ones that are already in travel, the maximum value should give the minimum number of trucks required for the operations. For the medium truck fleet size, this is a little more difficult because the transit time is not a single day and is function of the destination FC (more specifically, the distance between the DC and FC as well as the truck speed). The code provided takes this into consideration by adding a dictionary with the locations of the FC and their associated transit days (calculated using the method presented in subsection 2.9.3). We obtain the following results: for the medium truck fleet, we need 93 units, for the small truck fleet, we will need 86 trucks. These results are slightly different from the ones determined in task 2.8, however they are relatively close. This validates the results from the previous task (comparing with the maximum number of trucks required for the 2033 period).

2.10 Estimation of the expected investments, operating cost, energy consumption, and greenhouse gas emission

We estimate the expected investments, operating cost, energy consumption, and greenhouse gas emissions of the FCs, DC, warehouse, and fleet.

In estimating expected investments of the FCs, DC, and warehouse, we utilize a cost per square foot measurement with construction costs of \$100 per square foot according to CBRE Industrial & Logistics reports. In estimating operating cost of the FCs, DC, and warehouse, we consider labor cost of \$20 per hour, lease costs \$5 per sq. ft per year, maintenance cost of 2% of the investment cost, and utility cost (electricity and water) \$3 per square feet per year according to Prologis Industrial Business Indicator Reports and U.S. Energy Information Administration (EIA) – Commercial Buildings Energy Consumption Survey (CBECS). The electricity cost is derived from a per square feet energy intensity rate 20 kWh per square feet per year (which is also used to estimate the energy consumption) and the average electricity rate, and the water cost is derived from a per square feet water usage and the average industrial water cost rate. In estimating greenhouse gas emissions of the FCs, DC, and warehouse, we utilize the estimated energy consumption with an electricity emission factor of 0.38 kg CO_2/kWh according to the U.S. EPA Emissions & Generation Resource Integrated Database.

In estimating the expected investments of the fleet, we consider the cost of a semi-truck to be \$200000, the cost of a heavy rigid truck to be \$120000, and the cost of a light rigid truck to be \$50000 according to the American Transportation Research Institute (ATRI). In estimating the operating cost of the fleet, we consider fuel, maintenance, and driver cost per mile, which aggregate to \$2 per mile according to the ATRI and the National Private Truck Council. In estimating the energy consumption of the fleet, we consider different distance per litter of different types of trucks, with semi-trucks having 45l/100km, heavy rigid trucks having 35l/100km, and light rigid trucks having 17.5l/100km according to the U.S. DOE Alternative Fuels Data Center. In estimating the greenhouse gas emission of the fleet, we utilize the estimated energy consumption with a diesel emission factor of 10.21 kg CO_2/gallon, according to the U.S. EPA SmartWay Program.

Table 15 shows all the factors we discussed above:

Factor	**Estimation**
Building Investment	\$100/sq. ft.
Building Operational Cost - Labor	\$20/worker hour
Building Operational Cost - Lease	\$5/sq. ft.
Building Operational Cost - Maintenance	2% of Building Investment
Building Operational Cost - Utility	\$3/sq. ft.
Building Energy Consumption	20kWh/sq. ft.
Building Emission	0.38 kg CO2/kWh
Fleet Investment - Semi	\$200000/truck
Fleet Investment - Heavy	\$120000/truck
Fleet Investment - Light	\$50000/truck
Fleet Operational Cost	\$2/mile
Fleet Energy Consumption - Semi	45 l/100km
Fleet Energy Consumption - Heavy	35 l/100km
Fleet Energy Consumption - Light	17.5 l/100km
Fleet Emission - Semi	1.1 kg CO2/km
Fleet Emission - Heavy	0.9 kg CO2/km
Fleet Emission - Light	0.45 kg CO2/km

Table 15: Task 2.10 Estimations

Combined with the building plans of Task 2.2, Task 2.3, and Task 2.6 and the fleet plan of Task 2.10, we derive the yearly expected investments, operating cost, energy consumption, and greenhouse gas emissions of the FCs, DC, warehouse, and fleet, which are shown below. We separate the fleet's energy consumption from the others, as the fleet consumes litters of fuel rather than kWh of electricity. By observing the graphs, all factors mostly increase as the year progresses. In yearly investments, FCs and the fleet are the major costs. This is because FCs are required to be built at multiple sites. The fleet also dominates in operational costs and especially in GHG emissions. These suggest the importance of smart fleet routings and building positioning to achieve cost efficiency and sustainability. Full details can be viewed in the file `:/Task 2/Task 2.10/task 2.10.ipynb`.

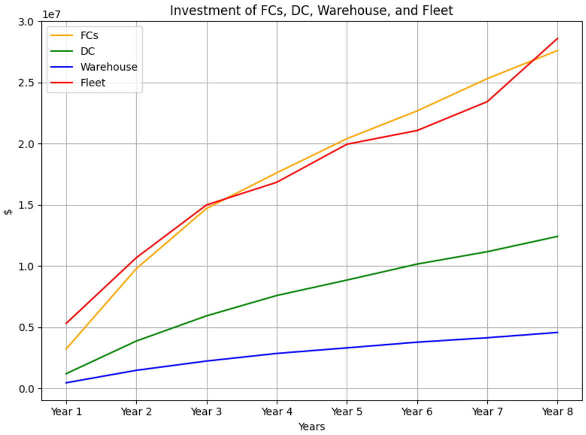


Figure 53: Yearly Investment of FCs, DC, Warehouse, and Fleet

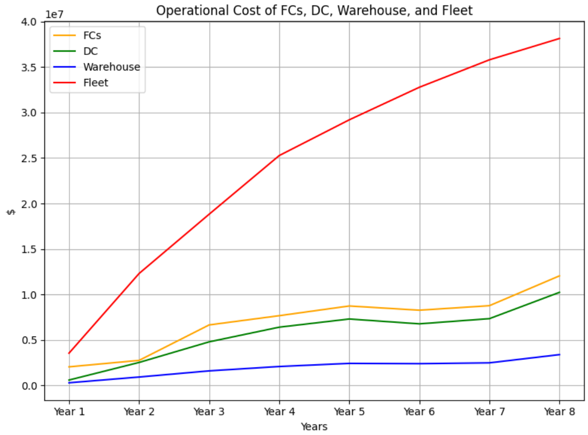


Figure 54: Yearly Operational Cost of FCs, DC, Warehouse, and Fleet

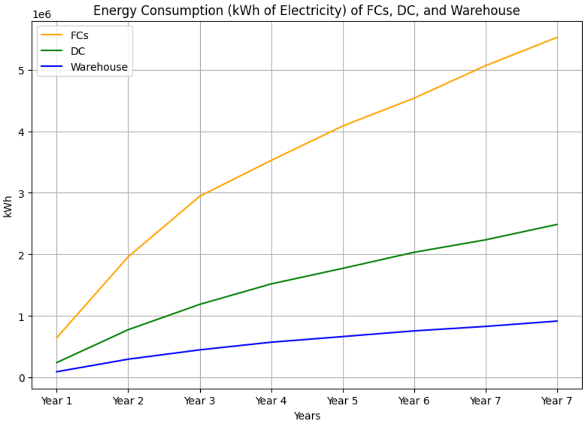


Figure 55: Yearly Energy Consumption of FCs, DC, and Warehouse

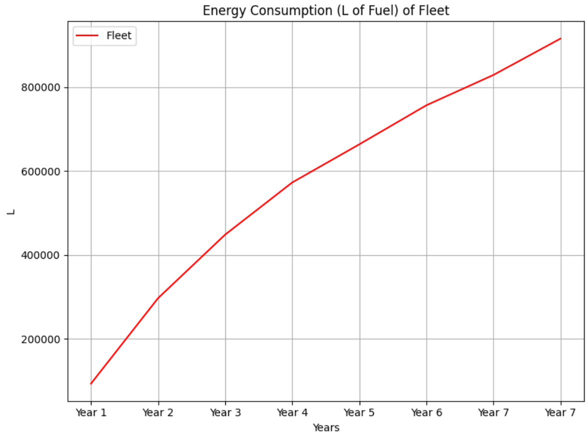


Figure 56: Yearly Energy Consumption of Fleet

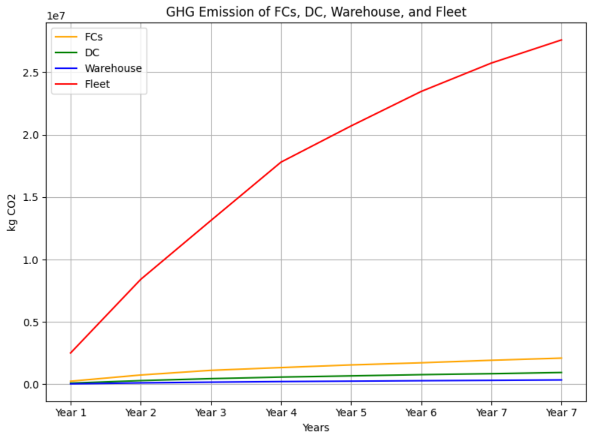


Figure 57: Yearly GHG Emission of FCs, DC, Warehouse, and Fleet

3 TASK 3: Critical Assessment

3.1 Strengths and weaknesses of our designed dedicated supply chain for BotWorld

The designed network efficiently reduces travel time between fulfillment centers and customers, which enhances delivery performance. The choice of fulfillment center locations considers nearness and demand density, which helps reduce operational costs. A significant strength is that the simulations provided robust demand forecasts, which can guide strategic decisions. Furthermore, having the Warehouse, the Plant and the DC in the same place helped saving costs in transportation. Having only one DC in Europe has allowed us to optimize this space as much as possible, having a single inventory management to take care of, allowing higher SKU density, better consolidation opportunities.

However, some weaknesses exist. The reliance on dense urban areas makes the design less adaptable to changes in rural demand. Optimization models used may have overlooked factors like traffic delays, which could impact real-world results.

Some inaccuracies may also remain when simulating the demand. We computed Monte Carlo simulations using the optimistic demand scenario, generating three different instances of the demand and taking the mean. In order to have a much more precise and robust demand prediction, we should consider at least 100 independent iteration of the demand and take the 1 st percentile as a benchmark for a 99% robust demand. This would be more precise, but also more time consuming, since iterating the demand over 8 years for 20 different products for +60 different locations takes over 2 hours

Similarly, in order to satisfy the delivery policy, we calculated we needed more drivers than trucks. However, labor is one of the most costly expenses in a company, and two or more drivers taking turns driving the same truck may not be very realistic (cf Physical Internet solutions).

Also, to simplify our calculations, we assumed that there were only single SKU (or product) pallets. This is not very realistic, since many logistic companies today tend to do pallet consolidation in order to save space in the trucks. This could help save a great number of trips where in some cases, the trucks would move almost empty.

3.2 Key findings and recommendations for BotWorld

The initial strategic market deployment roadmap might not have been the most profitable as we have to put FC on small Islands like Malta where the demand is not as high as in other areas like Brussels.
The strategy in itself (same day delivery) can prove to be extremely costly. In some instances, LTL shipments may sometimes be composed of a unique pallet. This reduces the pipeline inventory but greatly increases the transportation costs as well as the overall greenhouse gaz emissions. In some cases, when distributing to far FCs with very few demand, it may be more relevant to do some consolidation beforehand. One way would be to identify the primary, secondary and tertiary markets in the different urban areas and to do *same day delivery*, *next day delivery*, etc.
As we have seen in task 2.3, the ideal size of the distribution center may vary throughout the years. We would recommend directly buying a vast facility and filling it up as the years go by. The plans proposed in task 2.3 for the DC (as well as the FC - plans not required) show that we needed to expand the facility once. Buying a small facility that can be expended later on may be the right solution for WorldBot as the most crucial part in supply chain implementation is demand forecasting.
The main point of concern is the demand forecasting for these types of products. Even though WorldBot is already implemented in Asia and the Americas, the cultural differences with Europe might skew the predictions. Furthermore, we implemented a entire supply chain network by assuming a 99% robustness. However the demand may not be that high and the company may find itself loosing a lot of money due to its long term liabilities contracted through its large investment in trucks, facility, goods and labor. This is why we would recommend a gradual investment and quarterly reassessments of the demand.

3.3 Impact on the supply chain design and performance if Europe had already achieved a mature comprehensive Physical Internet implementation across its entire territory

We consider the impact of mature Physical Internet implementation on our supply chain design: 1) adoption of smart world-standard PI containers, 2) development of distributed multi-segment intermodal transport, and 3) evolution of an open supply web.

BotWorld's storage and logistics network will observe a huge improvemtn by adopting PI containers. With modularized PI containers, shipments will take less space than the current one-size-fits-all pallet-oriented story systems. As our previous tasks show, BotWorld's products are different in dimensions, resulting in notable space wastes in pallets, which are sometimes product-specific. PI containers will resolve this issue by having modularized and customized fits for all shipments, reducing space waste in both the FCs, DC, and warehouse storage systems. By requiring less space, PI containers will also reduce the number of fleets needed at all levels. PI containers are also stackable, which the current pallets are not, which will require even less space and trucks. Furthermore, PI containers are easily handled and transported with tracking and safety-related functions. These innovative designs will reduce product damages and losses that are not considered in this casework.

BotWorld's current logistics network is end-to-end and only involves trucks of different types. Besides, only one destination's shipments are transported by a truck. The costs and emission analyses of Task 2.10 show the drawbacks of this transportation system. Through a distributed multi-segment intermodal transportation network, shipments of different destinations will share truckload spaces in the same truck, delivering each shipment group through a multi-segmented network. Moreover, we will further reduce cost and emission levels by relying on multimodal transportation, such as railroad and liner shipping, with smart facility locations around multimodal hubs. Such a network will result in many transshipment requirements, challenging the current pallet-oriented storage system. However, PI containers will make this process easy and efficient, as stated above.

Finally, an open supply web will fundamentally change BotWorld's entire system. For example, as the prompt and simulation results show, BotWorld's demand is seasonal, and not all storage spaces will be utilized throughout the year. Therefore, in an open supply web, BotWorld can share its empty storage space with other companies or organizations that cannot afford a fixed facility. While receiving profits from doing so, BotWorld also enables other smaller players to facilitate their supply chain needs. Another example is that BotWorld can choose not to rely on its logistics network. Instead, it can depend on another or many other logistics providers. In such an open supply web, BotWorld does not need to sign a fixed contract with crazy terms and high costs like the current scenario. Rather, it will employ a collaboration strategy that is dynamic and real-time.

3.4 Key learnings from realizing this casework

This casework has helped us discover tools to compute road travel distances using python libraries. We could even have computed travel times that would consider congestion if we wanted to pay for other python modules. Using this tool, we have been able to efficiently determine where to put the FC, which is something essential if we are ever to help a new business implementing. We improved our demand forecasting skills using Monte Carlo simulations. We also learned how to compute complex schedules with multi parameters (yearly truck scheduling using Python).
We further refined our skills in Excel, Latex and AutoCad.
This casework clearly showcased the limits of same/next day delivery in today's supply chain networks and has instilled ideas of improvement and change (using ideas form the Physical Internet for instance).