40.002 Optimisation

Optimizing the Fisheries Industry in the Philippines

Prepared by:

Blauta Isaiah Rafael Tugano, 1007211

Harrish Rahul Venkatraman, 1006937

Lim Kyu Ha, 1006615

Nathan Ansel, 1007492

Course Instructors:

Karthik Natarajan

Antonios Varvitsiotis

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

1 Project Background

Protein deficiency and poverty in the Philippines is a an alarming issue, especially among children and pregnant women. Limited access to protein-rich food contributes to malnutrition and stunted growth, affecting health and development while also worsening poverty in the long term. To tackle this issue, citizens of the Philippines have focused on catching fish as one of the ways to provide a more affordable source of protein. However, overfishing and inefficient resource allocation have compromised the steady and reliable supply of fish—and with it, protein—while also worsening the economic security of the people.

Thus, this project strives to create a Mixed Integer Linear Program (MILP) to address this issue. Our MILP is designed to find an optimal strategy to catch and distribute fish among the people so as to maximize the overall protein distribution while minimizing all associated costs.

2 Formulating the Model

The Philippines is one of the world's biggest archipelagos: it is made up of multiple islands interconnected by sea. As such, the Philippines is blessed with abundant natural resources from the sea, namely fish. However, a major challenge is posed in that this fish is not directly accessible to the people. For an ordinary person to consume fish, fish must be distributed through an intricate supply chain involving, from the very beginning, a fishing ground, and then a processing facility, and finally a city from where a person can directly buy fish. Consider the following diagram to illustrate this supply chain.

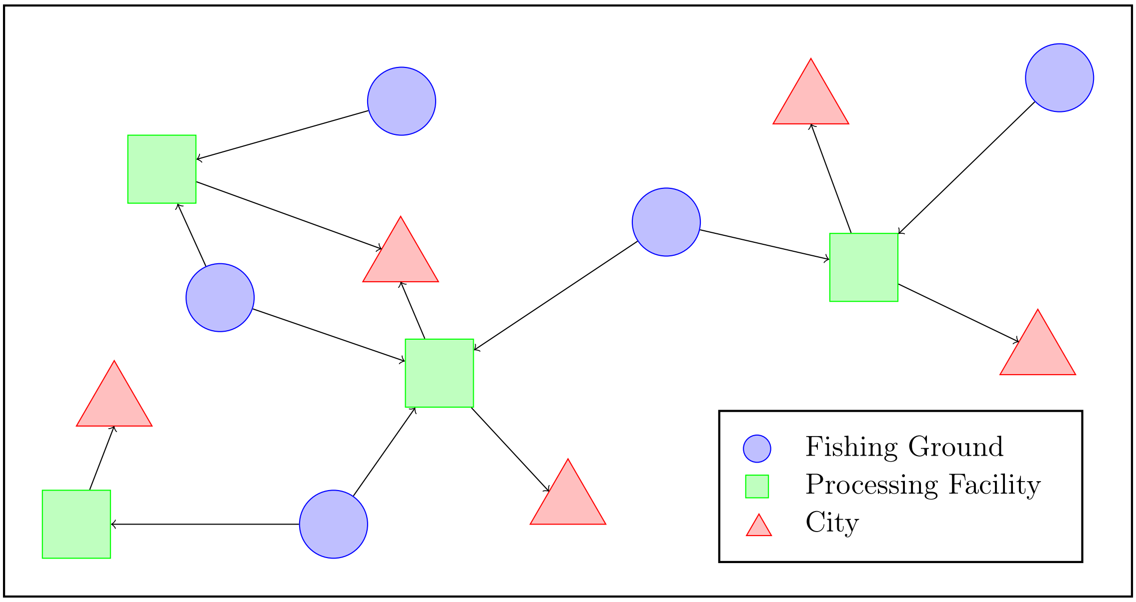

Figure 1: Problem Illustration, with an example of the fish distribution

A major challenge faced in the fisheries industry is thus deciding how to distribute fish in the most logistically efficient manner. Evidently, there are various geographical obstacles that separate the ordinary fish consumer from the fishing grounds—be it sea or land or even other obstacles—and hence the goal of this project would to be figure out an optimal strategy to distribute the most amount of fish while incurring the least amount of cost possible.

The problem thus centers around: (1) finding the optimal number of fish to be delivered from a fishing ground to a processing facility, (2) finding the optimal number of fish to be delivered from a processing facility to a city, (3) finding the optimal combination of facilities to be opened to ensure adequate fish processing capacity, and finally (4) finding the optimal amount of government subsidies required to provide access to fish consumption for the less fortunate—all while ensuring certain constraints are met.

Indeed, this problem is a "Multi-Stage" Capacitated Facility Location Problem: each processing facility is a client to the fishing grounds, and each city is a client to the processing facilities.

2.1 Decision Variables

The model consist of the following decision variables:

- x_{fgi} : Amount of fish f, in kg, to be caught from fishing ground g and sent for processing in facility i in a year.
- y_{fij} : Amount of fish f, in kg, to be sent from facility i to city j in a year.
- z_i : Boolean decision variable to indicate whether or not to open facility i in a year. $z_i = 1$ if facility i is open, 0 otherwise.
- s_j : Amount of government subsidy for city j in a year, in Philippine Peso (PHP).

2.2 Objective Function

Our objective function consists of several components.

Total Protein Value

We want the fisheries industry to benefit the people by providing an adequate supply of protein on a periodical basis (for the sake of this project, this periodical basis is 1 year). As such, we want to maximize the people's protein intake (to a reasonable degree, subject to certain health constraints). The people's overall protein consumption can be computed by the following equation:

$$Z_1 = \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} \sum_{j=1}^{|J|} v_f \cdot y_{fij}$$

where v_f denote the protein value, in grams per kilogram, of the cooked contents of fish f. $|F|$, $|I|$, and $|J|$ denote the total number of unique fish species, processing facilities, and cities.

Total Fishing Cost

While it is important to catch many fish, it is equally important to do so in the most efficient manner. As such, we need to minimize the total cost associated with catching and transporting fish from the fishing grounds to facilities. This total fishing cost is represented by the following equation:

$$Z_2 = \sum_{f=1}^{|F|} \sum_{g=1}^{|G|} \sum_{i=1}^{|I|} c_{fgi} \cdot x_{fgi}$$

where c_{fgi} denote the cost of catching and transporting 1 kg of fish f from fishing ground g to facility i. The cost here is interpreted as the total distance travelled from fishing ground g to facility i. $|G|$ denote the total number of unique fishing grounds.

Total Distribution Cost

Similarly, we are interested to minimize the total cost associated with distributing fish from the fishing grounds to the facilities. This total cost is represented by the following equation:

$$Z_3 = \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} \sum_{j=1}^{|J|} d_{fij} \cdot y_{fij} + \sum_{i=1}^{|I|} o_i \cdot z_i$$

where d_{fij} denote the cost of distributing 1 kg of fish f from processing facility i to city j and o_i denote the yearly operational cost of processing facility i, in PHP. The cost d_{fij} is interpreted as the total distance travelled from facility i to city j.

Total Cost of Subsidies

Finally, the government may need to provide subsidies into the fisheries industries to ensure adequate access to fish consumption for the people. However, a central part of the problem is to decide how to allocate the subsidies such that only those who really need it will receive it. This can be done by minimizing the total cost of subsidies (subject to certain constraints to be discussed below). The following equation describes the total amount of subsidies required.

$$Z_4 = \sum_{j=1}^{|J|} s_j$$

The subsidies in this context refer to the amount of money given directly to the people to help them purchase fish or fish-derived food.

Objective Function

Our objective function can thus be summarised as follows.

$$\max Z = M \cdot Z_1 - Z_2 - Z_3 - Z_4$$

where M is a very large number which enables the model to prioritize maximizing the total protein distributed. Assuming $M = 10^5$, then we have:

$$\begin{aligned} \max Z = &\left(10^5 \cdot \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} \sum_{j=1}^{|J|} v_f \cdot y_{fij}\right) - \left(\sum_{f=1}^{|F|} \sum_{g=1}^{|G|} \sum_{i=1}^{|I|} c_{fgi} \cdot x_{fgi}\right) \\ &- \left(\sum_{f=1}^{|F|} \sum_{i=1}^{|I|} \sum_{j=1}^{|J|} d_{fij} \cdot y_{fij} + \sum_{i=1}^{|I|} o_i \cdot z_i\right) - \left(\sum_{j=1}^{|J|} s_j\right) \end{aligned}$$

Further discussion regarding this approach is given in section 4.2

2.3 Constraints

There are several constraints which restrict us from infinitely improving our optimal solution.

Non-negativity Constraint

Firstly, we have several non-negativity constraints.

$$\begin{aligned} x_{fgi} &\geq 0 && \forall f = 1, \ldots, |F|,\ \forall g = 1, \ldots, |G|,\ \forall i = 1, \ldots, |I| \\ y_{fij} &\geq 0 && \forall f = 1, \ldots, |F|,\ \forall i = 1, \ldots, |I|,\ \forall j = 1, \ldots, |J| \\ s_j &\geq 0 && \forall j = 1, \ldots, |J| \end{aligned}$$

These constraints are necessary as neither x_{fgi}, y_{fij}, nor s_j can be negative in real life. Thus the constraint ensures the model will not produce any unrealistic optimal solution.

Integrality Constraint

Following the definition of the decision variable z_i, we have the integrality constraint:

$$z_i \in \{0, 1\} \qquad \forall i = 1, \ldots, |I|$$

where $z_i = 1$ if facility i is open, and $z_i = 0$ otherwise.

Fishing Restriction Constraint

To ensure long term environmental sustainability, the government implements a restriction to the amount of fish that can be caught. For every fishing ground g, there can only be a maximum of $R_{\text{max, } fg}$ kg of fish f caught every year, thus giving the constraint:

$$\sum_{i=1}^{|I|} x_{fgi} \leq R_{\text{max, } fg} \qquad \forall f = 1, \ldots, |F|, \ \forall g = 1, \ldots, |G|$$

$R_{\text{max, } fg}$ is calculated in accordance to the Law of Philippines.

Processing Capacity Constraint

Each processing facility has a certain capacity restriction due to logistics limitation (limited number of workers, storage space, machinery, etc.). As such, each processing facility i is constrained to a certain maximum processing capacity of $P_{\text{max, } i}$ (measured in kg) every year.

$$\sum_{f=1}^{|F|} \sum_{j=1}^{|J|} y_{fij} \leq P_{\text{max, } i} \qquad \forall i = 1, \ldots, |I|$$

Protein Consumption Constraint

We need to ensure that the people will receive neither too little nor too much protein on every periodic basis. As such, for every city j, the people's total protein consumption must be within the following range as recommended by health experts:

$$1/3 \cdot \rho_j \cdot M_1 \leq \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} v_f \cdot y_{fij} \leq \rho_j \cdot M_2 \qquad \forall j = 1, \ldots, |J|$$

where M_1 and M_2 denote the minimum and maximum yearly protein consumption per person (measured in grams of protein per kilogram of body weight); and ρ_j denote the population of city j. Here, we selected $1/3 \cdot \rho_j \cdot M_1$ as the lower bound for protein consumption because a human needs to take protein from a diverse range of sources, not just fish. Thus, 1/3 is selected as a conservative lower bound: it ensures that Filipinos can consume protein from a wide variety of sources while requiring them to consume at least 1/3 of their protein from fish. While 1/3 is an arbitrary value that is modifiable, this constraint makes the model more realistic nonetheless.

Using data from Harvard Health Publishing, we calculated M_1 and M_2 to be approximately 17,520 and 43,800 and thus:

$$5,840 \cdot \rho_j \leq \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} v_f \cdot y_{fij} \leq 43,800 \cdot \rho_j \qquad \forall j = 1, \ldots, |J|$$

Distribution Capacity Constraint

Each facility cannot distribute more fish than what it receives. As such, we have the constraint:

$$\sum_{j=1}^{|J|} y_{fij} \leq \sum_{g=1}^{|G|} x_{fgi} \qquad \forall f = 1, \ldots, |F|, \ \forall i = 1, \ldots, |I|$$

This constraint ensures that the sum of fish f distributed to all cities from facility j does not exceed the sum of fish f received by facility j from all fishing grounds.

Budget Constraint

It is impossible for people to buy more fish than what they can afford. As such, we have the constraint:

$$\sum_{f=1}^{|F|} \sum_{i=1}^{|I|} c_f \cdot y_{fij} \leq (15\%) \cdot \rho_j \cdot b_j + s_j \qquad \forall j = 1, \ldots, |J|$$

where c_f denote the market price of buying 1 kg of fish f (in PHP); ρ_j denote the population of city j; and b_j refer to the average GDP per capita in city j (in PHP).

Assuming that a person would reasonably spend at most 15% of their income on fish every year, we selected $(15\%) \cdot \rho_j \cdot b_j$ as a reasonable upper bound for the initial budget of all people in city j. As subsidies are also allocated by the government to each city, the final budget for a city j becomes $(15\%) \cdot \rho_j \cdot b_j + s_j$

Fish Diversity Constraint

Another critical aspect of the problem is to ensure the diversity of fish sold in the market. While some fish may appear to be economically more profitable to sell compared to others, it is not wise to apply so much ecological pressure into a select group of fish species. Rather, it would be best, both for the environment and the people's health, to ensure a diverse range of fish species being consumed. Hence, we have the following constraint which sets a minimum and maximum proportion of fish species distributed in a city.

$$(2.5\%) \cdot \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} y_{fij} \leq \sum_{i=1}^{|I|} y_{fij} \leq (25\%) \cdot \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} y_{fij} \qquad \forall f = 1, \ldots, |F|, \ \forall j = 1, \ldots, |J|$$

That is, every fish species f must contribute to at least 2.5% and at most 25% to the total fish consumption in a city j. The numbers 2.5% and 25% were selected arbitrarily; these numbers can be modified, if need be, according to recommendations made by environmental experts.

Fishing Ground Diversity Constraint

Similarly, it is not wise to apply so much ecological pressure into a select group of fishing grounds. As such, to ensure a diverse utilization of fishing grounds, we have the constraint:

$$(2.5\%) \cdot \sum_{f=1}^{|F|} \sum_{g=1}^{|G|} \sum_{i=1}^{|I|} x_{fgi} \leq \sum_{f=1}^{|F|} \sum_{i=1}^{|I|} x_{fgi} \leq (25\%) \cdot \sum_{f=1}^{|F|} \sum_{g=1}^{|G|} \sum_{i=1}^{|I|} x_{fgi} \qquad \forall g = 1, \ldots, |G|$$

Facility Opening Constraint

Finally, another equally critical aspect of the problem is to decide which combination of facilities to open every year. If we decide to open a certain processing facility i, this means that the facility can process a certain amount of fish subject to the constraints previously discussed. However, if we decide *not* to open facility i, this means that the facility cannot process any fish at all. Thus, it can neither (1) receive any fish from the fishing grounds, nor can it (2) send any processed fish to the cities.

To satisfy (1), we have the following relationship for every processing facility i:

$$z_i = 0 \implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq 0 \qquad \forall i = 1,\ldots,|I|$$

$$z_i = 1 \implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

where z_i is a binary decision variable which takes value 1 when facility i is open and 0 otherwise. This relationship can be expressed by the following constraint.

$$\sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} z_i \cdot R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

Indeed, when $z_i = 0$, we restrict x_{fgi} from any positive values. This means that no fish can be sent to facility i from any fishing ground as x_{fgi} are restricted to non-negative values by the non-negativity constraint.

$$z_i = 0 \implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} 0 \cdot R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

$$\implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq 0 \qquad \forall i = 1,\ldots,|I|$$

On the other hand, when $z_i = 1$, the constraint reduces to an aggregated formulation of the Fishing Restriction Constraint:

$$z_i = 1 \implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} 1 \cdot R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

$$\implies \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

The above inequality is true for the following reason. Since we have

$$\sum_{i=1}^{|I|} x_{fgi} \leq R_{\text{max},\ fg} \quad \forall f, g \qquad \text{(Fishing Restriction Constraint)}$$

$$x_{fgi} \geq 0 \quad \forall f, g, i \qquad \text{(Non-negativity Constraint)}$$

It also follows that:

$$x_{fgi} \leq R_{\text{max},\ fg} \quad \forall f, g, i$$

If we sum the above inequality for all f and g, we get the aggregated formulation for the Fishing Restriction Constraint.

$$\sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} R_{\text{max},\ fg} \qquad \forall i = 1,\ldots,|I|$$

This constraint is consistent with the inequality derived earlier. That is, when $z_i = 1$, the constraint allows us to send all types of fish from all fishing grounds to facility i so long as the Fishing Restriction Constraint is not violated. This explains why the constraint is able to capture the relationship between z_i and x_{fgi}.

It is important to note that the inequality $\sum_{f=1}^{|F|}\sum_{g=1}^{|G|} x_{fgi} \leq \sum_{f=1}^{|F|}\sum_{g=1}^{|G|} z_i R_{\text{max, } fg}$, $\forall i$ provides a stronger formulation than (say) $x_{fgi} \leq z_i R_{\text{max, } fg}$, $\forall f, g, i$, as the former requires a total of only i constraints whereas the latter requires $f \times g \times i$ constraints. In practice, while both formulations are valid, our formulation will enable the solver to terminate faster.

Meanwhile, to satisfy (2), we have the following relationship for every facility i:

$$z_i = 0 \implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq 0 \qquad \forall i = 1, \ldots, |I|$$

$$z_i = 1 \implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq P_{\text{max, } i} \qquad \forall i = 1, \ldots, |I|$$

To represent this relationship, we introduce the constraint:

$$\sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq z_i \cdot P_{\text{max, } i} \qquad \forall i = 1, \ldots, |I|$$

Indeed, when $z_i = 0$, the constraint restricts the i-th facility from distributing any fish.

$$\begin{aligned} z_i = 0 &\implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq 0 \cdot P_{\text{max, } i} && \forall i = 1, \ldots, |I| \\ &\implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq 0 && \forall i = 1, \ldots, |I| \end{aligned}$$

On the other hand, when $z_i = 1$, the constraint reduces into the Processing Capacity Constraint:

$$\begin{aligned} z_i = 1 &\implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq 1 \cdot P_{\text{max, } i} && \forall i = 1, \ldots, |I| \\ &\implies \sum_{f=1}^{|F|}\sum_{j=1}^{|J|} y_{fij} \leq P_{\text{max, } i} && \forall i = 1, \ldots, |I| \end{aligned}$$

Thus, this constraint correctly captures the relationship between z_i and y_{fij}.

2.4 Modelling Assumptions

The model relies on the following assumptions:

- All fish distributed to a city are purchased completely by the people; there are no leftovers. In other words, it is assumed that the market demand for fish f in a city j is equal to the sum of fish f delivered to city j.
- For every city, the GDP per capita is representative of a person's yearly purchasing power.
- All the parameters and data (such as processing capacity) are constant and known for the whole year. In reality, the data may vary over time due to various factors such as seasonality and consumer preferences.
- The relationship between the decision variables are linear.
- There are no economies of scale in the fisheries industry.

3 Implementing the Model

In this section, we explain how we implemented the model using Julia.

3.1 Data Collection

Data collection is a critical part of the implementation. To retrieve the most realistic data, our team collected data from authoritative sources such as the Philippines Fisheries Development Authority (PFDA) and the Bureau of Fisheries and Aquatic Resources (BFAR). Where data is unavailable, we estimated or randomized the data using certain techniques explained below. All data sources used can also be found in the references section at the end of this report.

In addition, to ensure coherency between different datasets, 2021 was selected as the year of the study. Thus, the datasets used will be from the year 2021 and, where unavailable, the dataset with the closest year to 2021 will be used.

Fish Data

Fish data is compiled from sources such as the Philippines Government website, authorized statistics website and trusted sites on food nutrition such as Statista. Some fish prices are extrapolated based on the trend of prices due to the unavailability of consistent, single source data. In total, the model considers 7 fish species: Milkfish, Tilapia, Galunggong, Yellowfin Tuna, Indian Mackarel, Skipjack Tuna, and Shrimp. If need be, the model can take more species, but for the sake of this implementation, only 7 of the most popular fish was selected.

Fishing Ground Data

The model considers a total of 16 fishing grounds which covers all seas in the Philippines (please check the appendix to see the names of all the fishing grounds). Data for these fishing grounds is readily available yearly as the Bureau of Fisheries and Aquatic Resources (BFAR) releases a yearly report for each region in the Philippines. This report also includes the various types of fish and aquaculture found in the fishing ground regions. For years 2016-2021, all relevant data was available in the form of a detailed report downloadable from the official website of BFAR. We then took the time to read and extract the data from the summarised table reports of the yearly release. We curated these numbers yearly and by region. However, for the year 2016, 2017, 2018, the Shrimp data was not representative per region, as such, we took the total number of Shrimp and extrapolated the data based on the years ahead.

To compute the distance between the fishing grounds and facilities, we collected the coordinates (latitude and longitude) of each fishing ground through Google Maps search. Afterwards, using the Haversine formula, the distance between each point was calculated in *Python*. To make the data more interesting, for every fish f, the fishing cost is randomized such that it fluctuates between 70% to 130% of the original fishing cost. This approach was also done when calculating the cost of transportation from a facility to a city.

Processing Facility Data

To compute the distance between a facility and a city, it is crucial to collect the location data of processing facilities. This data was mostly acquired from the Philippines Fisheries Development Authority (PFDA), with some exception where it was not listed. Google Maps search was done for those anomalies to locate the fish processing facilities for each respective region. The maximum processing capacity of each facility was also taken from PFDA. Finally, the average income within the city or region of a facility was taken into account and multiplied with the total number of people employed in the fishing industry to come up with the annual operating cost of the facilities. This data was taken from Payscale. Overall, the model considers a total of 56 processing facilities distributed across different islands in the Philippines.

Cities

The model considers 100 cities with the highest population in the Philippines. To compute the distance between a facility and city, the coordinates data for the cities was collected from the SimpleMaps website. The website also provides relevant information regarding the population of the city and the administrative status.

Due to unavailable data, the GDP per capita in each city was estimated using the city's administrative status. The higher the administrative status, the higher the GDP per capita in this city. The precise values of the GDP per capita in each city was computed using the random function on *Excel*, but the result was such that each city's GDP per capita is within a reasonable range from the national GDP per capita (between 80% to 130% of the national GDP per capita).

Location Mapping

For visualisation purposes, we plotted all the fishing grounds, processing facilities, and cities using Google Maps. The interactive map can be accessed here.

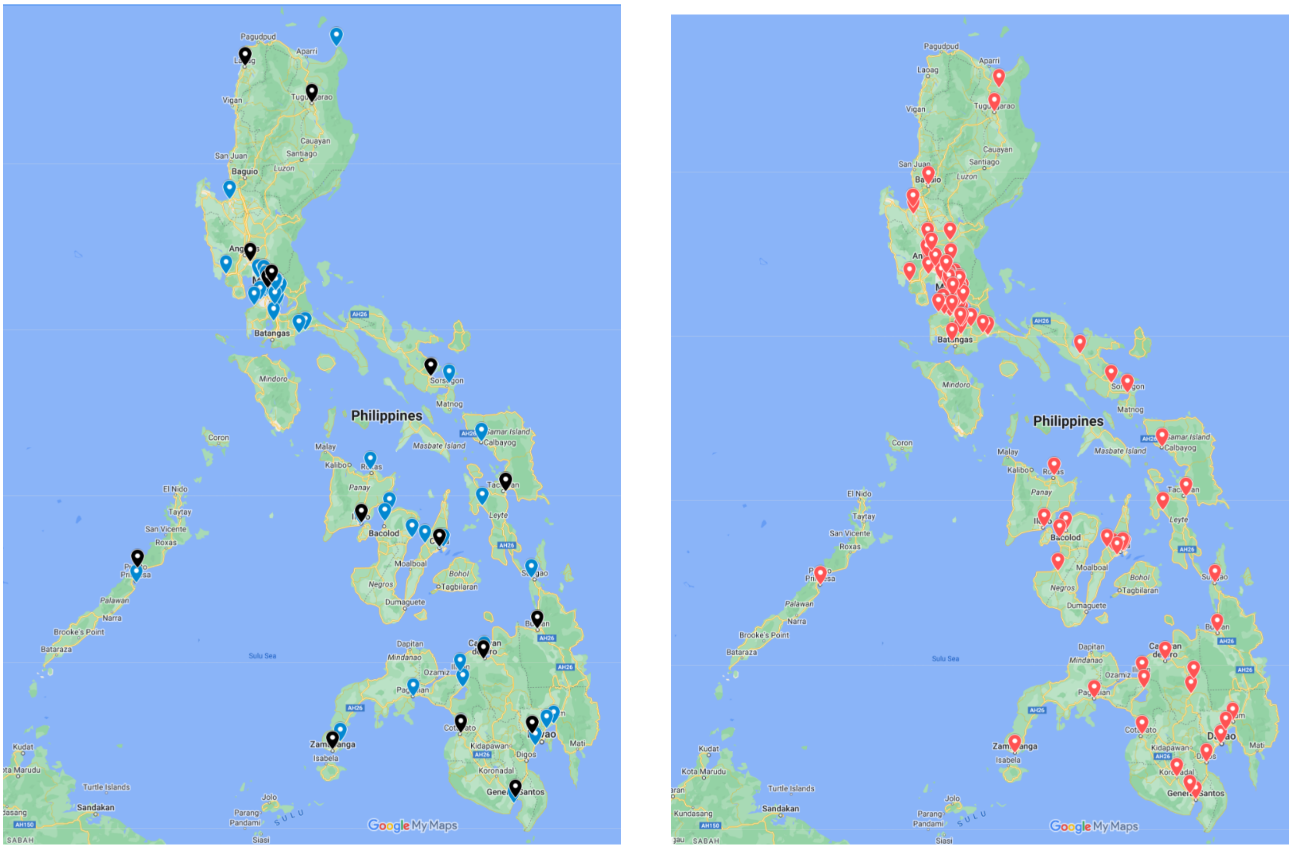

Figure 2: Map of all Fishing Grounds (black), Processing Facilities (blue), and Cities (red)

3.2 Julia and Results

The model was solved using Julia, particularly by using JuMP and GLPK as the solver. The full code for the problem can be seen in the appendix. After implementing the model, an optimal solution was found with optimal objective value of $Z^* = 1.0228604232602997 \times 10^{17}$. Due to space limitations, the optimal solutions were attached in Appendix A.

The optimal solution makes use of a diverse range of fish, processing facilities, and fishing grounds to maximize the objective function while also fulfilling all constraints. This makes it more practical for policymakers to implement any optimal solution produced by our model. Additionally, it is also interesting to note that $z_i^* = 1,\ \forall i$. This suggests that the opening of the facilities considered in the model are critical to successfully distribute fish across the Philippines. Equally interestingly is $s_j^* = 0,\ \forall j$, which indicates that the government does not require to provide any further subsidies to ensure adequate fish distribution to a city.

All in all, the fact that an optimal solution exists suggests that it is **possible** to maximize protein distribution in the Philippines through fish—especially without violating critical constraints such as those concerning sustainability and the environment. This poses an optimistic conclusion: the Philippines can leverage fish to address their protein needs while also ensuring environmental sustainability.

4 Evaluation

4.1 Strengths

Firstly, a notable strength of the model is that it takes various constraints into consideration, be it environmental constraints, circumstantial constraints, or budgetary constraints. Furthermore, these constraints did not over-restrict the feasible region, allowing for an optimal solution to be found. All in all, these constraints contribute to making the model more realistic, allowing for any optimal solution to be implemented more easily by policymakers.

Secondly, the model successfully combines two objectives: maximizing protein distribution through fish, and minimizing the total cost associated with all distribution efforts. In reality, both of these objectives are crucial and as such the model was intentionally designed to incorporate these objectives into the problem.

Thirdly, the model incorporates some interactions between different decision variables. This was captured via the Distribution Capacity Constraint, the Facility Opening Constraint, and the Budget Constraint. These constraints shape not only a more interesting problem but a more realistic one too.

4.2 Limitations

The model uses an approach somewhat similar to the Big M method, where the component Z_1 in the objective function is multiplied with a very large number $M = 10^5$. Here, M serves as a penalty for not maximizing the total protein value distributed (thus the model is "forced" to find an optimal solution where the total protein distributed is high). However, this approach may cause several issues. Firstly, introducing M may introduce numerical instabilities in the model, making it resource-consuming for a solver to terminate. Secondly, we selected an arbitrary value of M due to the constraints of the study. However, selecting an arbitrary value of M may pose a problem as different values of M can produce different optimal solutions. Thus, further work must be done to determine the right value of M.

5 Conclusion

All in all, our model is a stepping stone to optimizing the fisheries industry in the Philippines. The model accounts for various objectives and constraints faced by those working in the fisheries industry and as such, within the assumptions of the model, the model can advise relevant policymakers in the Philippines how to best develop the fisheries industry. Indeed, the optimal solutions found in this study suggests that the Philippines can leverage fish to address their protein needs—all while ensuring no constraint is violated. Thus, with strategic management and development, the fisheries industry can emerge as a transformative solution to the Philippines' enduring challenges.

References

Fish Data

Republic of Philippines. (n.d.). Fish and fishery product price monitoring service. https://www.bfar.da.gov.ph/market-development-services/fish-and-fishery-product-price-monitoring-service/

Published by Statista Research Department, & 14, S. (2023, September 14). Philippines: Domestic retail price of tilapia 2022. Statista. https://www.statista.com/statistics/1048036/domestic-retail-price-tilapia-philippines/

Published by Statista Research Department, & 14, S. (2023a, September 14). Philippines: Domestic retail price of milkfish 2022. Statista. https://www.statista.com/statistics/1047867/domestic-retail-price-bangus-milkfish-philippines/

Published by Statista Research Department, & 14, S. (2023b, September 14). Philippines: Domestic retail price of roundscad 2022. Statista. https://www.statista.com/statistics/1048032/domestic-retail-price-galunggong-mackerel-scad-philippines/

Price-monitoring-september-6-2022.PDF. (n.d.). https://www.da.gov.ph/wp-content/uploads/2022/09/Price-Monitoring-September-6-2022.pdf

Processing Facility Data

Maitem, C. E. and J. (2022, August 14). *In Philippines' tuna capital, fishermen are reeling from high fuel prices*. Benar News. https://www.benarnews.org/english/news/philippine/philippines-commercial-tuna-fishing-08122022180752.html

General Santos City, Philippines salary | payscale. (n.d.-a). https://www.payscale.com/research/PH/Location=General-Santos-City/Salary

Gan, M., & Rotz, M. V. (2017). Development For Whom? How Navotas Fisherfolk Resist the Displacement of Their People and Livelihood. Philippines; IBON INTERNATIONAL.

Agriculture and Fisheries. investiniloilo.ph. (2021, May 28). https://investiniloilo.ph/agriculture-and-fisheries/

Office, D. P. (2022, March 7). *From PHILSTAR: Lucena Fishport rehab 90% complete*. Official Portal of the Department of Agriculture. https://www.da.gov.ph/from-philstar-lucena-fishport-rehab-90-complete/

Narvaez, T. (2020, July 16). *Assessment of the industry-level impacts of the closed fishing season policy for sardines in Zamboanga Peninsula, Philippines*. FFTC Agricultural Policy Platform (FFTC-AP). https://ap.fftc.org.tw/article/1242

Fisheries situation in Davao de Oro January – March 2023 | Philippine Statistics Authority Region XI. (2023, September 7). https://rsso11.psa.gov.ph/content/fisheries-situation-davao-de-oro-january-march-2023

Institute, E. E. R. (n.d.). *Average salary in Davao, Philippines*. Salary Expert. https://www.salaryexpert.com/salary/area/philippines/davao

Sotelo, Y. (2016, November 10). *SUAL maintains status as top "bangus" producer*. INQUIRER.net. https://newsinfo.inquirer.net/842712/sual-maintains-status-as-top-bangus-producer

Fishing Ground Data

P. Velza, J. F., Ibañez, R., & Mahawan, A. (2022). *International Journal of Multidisciplinary: Applied Business and Education Research*. Impact Assessment of Aqua Silviculture of Milagros and Placer, Masbate. http://dx.doi.org/10.11594/ijmaber.03.09.17

Labayo, C. (2022, May 27). *Perceptions of fishermen households on a community-based Coastal Resource Management Area: The case of asid gulf, Masbate, Philippines*. Academia.edu. https://www.academia.edu/79489971/Perceptions_of_Fishermen_Households_on_a_Community_based_Coastal_Resource_Management_Area_The_Case_of_Asid_Gulf_Masbate_Philippines?source=swp_shares

Chong, K. C., Lizarondo, M. S., Holazo, V. F., & Smith, I. R. (1982). *Inputs as Related to Output in Milkfish Production in the Philippines.*

Philippines Department of Agriculture. (2021). *Philippine Fisheries Profile 2021 - BFAR*. Philippine Fisheries Profile.
https://www.bfar.da.gov.ph/wp-content/uploads/2022/11/2021-Fisheries-Profile-FINAL-FILE.pdf

Philippines Department of Agriculture. (2020). *Philippine Fisheries Profile 2020 - BFAR*. 2020 Fisheries Profile.
https://www.bfar.da.gov.ph/wp-content/uploads/2022/02/2020-Fisheries-Profile-Final.pdf

Philippines Department of Agriculture. (2019). *Philippine Fisheries Profile 2019 - BFAR*. Philippine Fisheries Profile 2019.
https://www.bfar.da.gov.ph/wp-content/uploads/2021/05/Philippine-Fisheries-Profile-2019.pdf

Philippines Department of Agriculture. (2018). *Philippine Fisheries Profile 2018 - BFAR*. Philippine Fisheries Profile 2018.
https://www.bfar.da.gov.ph/wp-content/uploads/2021/05/Philippine-Fisheries-Profile-2018.pdf

Philippines Department of Agriculture. (2017). *Philippine Fisheries Profile 2017 - BFAR*. Philippine Fisheries Profile 2017.
https://www.bfar.da.gov.ph/wp-content/uploads/2021/05/Philippine-Fisheries-Profile-2017.pdf

Philippines Department of Agriculture. (2016). *Philippine-Fisheries-Profile-2016.pdf - BFAR*. Philippine Fisheries Profile 2016. https://www.bfar.da.gov.ph/wp-content/uploads/2021/05/Philippine-Fisheries-Profile-2016.pdf

Cities Data

Philippines Cities Database. simplemaps. (n.d.). https://simplemaps.com/data/ph-cities

Other

Mean height and weight of Filipino adults by region | download scientific diagram. (n.d.-b).
https://www.researchgate.net/figure/Mean-Height-and-Weight-of-Filipino-Adults-by-Region_fig1_340685263

How much protein do you need every day?. Harvard Health. (2023, June 22).
https://www.health.harvard.edu/blog/how-much-protein-do-you-need-every-day-201506188096

When it comes to protein, how much is too much?. Harvard Health. (2020, March 30).
https://www.health.harvard.edu/nutrition/when-it-comes-to-protein-how-much-is-too-much

Appendix A - Results

The following table lists all x^*_{fgi} and y^*_{fij} which does not take zero as optimal value. This means that the rest of x^*_{fgi} and y^*_{fij} which are not included in this table equal 0.

Meanwhile, for z^*_i and s^*_j, we have:

$$z^*_i = 1 \quad \forall i = 1, \ldots, |I|$$
$$s^*_j = 0 \quad \forall j = 1, \ldots, |J|$$

Variable	Value	Variable	Value	Variable	Value
x[1, 1, 3]	30900000.00	x[2, 1, 1]	4570000	x[3, 1, 1]	3270000.00
x[1, 1, 8]	7980000.00	x[2, 1, 3]	36600000	x[3, 1, 3]	79600000.00
x[1, 1, 12]	47400000.00	x[2, 1, 7]	8310000	x[3, 1, 8]	1940000.00
x[1, 1, 13]	11900000.00	x[2, 1, 8]	7980000	x[3, 1, 12]	65800000.00
x[1, 1, 15]	6290000.00	x[2, 1, 12]	20800000	x[3, 1, 13]	8300000.00
x[1, 1, 16]	53200000.00	x[2, 1, 13]	22400000	x[3, 1, 16]	39400000.00
x[1, 1, 24]	7490000.00	x[2, 1, 15]	36000000	x[3, 1, 19]	2270000.00
x[1, 1, 27]	20400000.00	x[2, 1, 16]	56300000	x[3, 1, 24]	9670000.00
x[1, 1, 53]	2320000.00	x[2, 1, 24]	19300000	x[3, 1, 27]	14300000.00
x[1, 2, 40]	104000000.00	x[2, 1, 27]	20400000	x[3, 1, 30]	47500000.00
x[1, 2, 48]	12900000.00	x[2, 1, 36]	104000000	x[3, 1, 36]	5750000.00
x[1, 5, 45]	11300000.00	x[2, 1, 46]	16700000	x[3, 1, 46]	6750000.00
x[1, 6, 32]	48500000.00	x[2, 1, 52]	26600000	x[3, 1, 52]	10500000.00
x[1, 6, 33]	22800000.00	x[2, 1, 53]	6330000	x[3, 1, 53]	4430000.00
x[1, 6, 34]	100000000.00	x[2, 2, 40]	9470000	x[3, 2, 40]	34100000.00
x[1, 6, 38]	1610000.00	x[2, 3, 29]	12000000	x[3, 2, 48]	32200000.00
x[1, 6, 50]	33800000.00	x[2, 3, 41]	22500000	x[3, 3, 29]	8370000.00
x[1, 6, 53]	4020000.00	x[2, 4, 50]	39900000	x[3, 3, 41]	17500000.00
x[1, 7, 1]	12900000.00	x[2, 5, 45]	68000000	x[3, 4, 50]	24600000.00
x[1, 7, 4]	7230000.00	x[2, 6, 33]	17600000	x[3, 5, 45]	7900000.00
x[1, 7, 18]	44300000.00	x[2, 7, 4]	4470000	x[3, 6, 33]	15000000.00
x[1, 7, 19]	39000000.00	x[2, 7, 6]	36500000	x[3, 7, 6]	28900000.00
x[1, 7, 20]	2900000.00	x[2, 7, 18]	50400000	x[3, 7, 18]	4570000.00
x[1, 7, 26]	15800000.00	x[2, 7, 19]	8170000	x[3, 7, 26]	44900000.00
x[1, 8, 28]	44300000.00	x[2, 7, 26]	15800000	x[3, 7, 30]	267658.00
x[1, 9, 31]	34600000.00	x[2, 7, 30]	33700000	x[3, 8, 28]	39300000.00
x[1, 10, 2]	42200000.00	x[2, 7, 34]	3570000	x[3, 9, 31]	1280000.00
x[1, 10, 14]	36000000.00	x[2, 8, 28]	12800000	x[3, 11, 35]	4050000.00
x[1, 10, 42]	5610000.00	x[2, 9, 10]	4180000	x[3, 11, 47]	7630000.00
x[1, 11, 35]	19900000.00	x[2, 9, 31]	47600000	x[3, 12, 39]	12000000.00
x[1, 11, 47]	986822.70	x[2, 9, 49]	9830000	x[3, 13, 43]	16100000.00
x[1, 12, 39]	7910000.00	x[2, 10, 2]	42200000	x[3, 14, 41]	10200000.00
x[1, 13, 5]	26900000.00	x[2, 10, 14]	31000000	x[3, 15, 51]	73900000.00
x[1, 13, 23]	7080000.00	x[2, 10, 22]	4940000		
x[1, 13, 43]	16000000.00	x[2, 10, 42]	5610000		
x[1, 14, 29]	7200000.00	x[2, 11, 35]	10400000		
x[1, 14, 41]	43200000.00	x[2, 11, 37]	4170000		
x[1, 15, 21]	16700000.00	x[2, 11, 47]	11500000		

Continued on next page

Variable	Value	Variable	Value	Variable	Value
x[1, 15, 51]	7140000.00	x[2, 11, 48]	18700000		
x[1, 16, 25]	10400000.00	x[2, 12, 39]	57600000		
		x[2, 13, 5]	9460000		
		x[2, 13, 43]	29400000		
		x[2, 14, 41]	17100000		
		x[2, 15, 21]	14400000		
		x[2, 15, 51]	20500000		
		x[2, 16, 25]	83400000		

Variable	Value	Variable	Value	Variable	Value
x[4, 1, 3]	4370000.00	x[5, 1, 3]	3440000.00	x[6, 1, 3]	3440000.00
x[4, 1, 52]	10300000.00	x[5, 1, 7]	383987.50	x[6, 1, 8]	798494.10
x[4, 2, 40]	19800000.00	x[5, 1, 8]	798494.10	x[6, 1, 9]	13200000.00
x[4, 3, 29]	12000000.00	x[5, 1, 9]	1640000.00	x[6, 1, 12]	4740000.00
x[4, 3, 41]	29500000.00	x[5, 1, 12]	4740000.00	x[6, 1, 13]	1190000.00
x[4, 4, 50]	37400000.00	x[5, 1, 13]	1190000.00	x[6, 1, 15]	629217.00
x[4, 5, 45]	11300000.00	x[5, 1, 15]	245229.50	x[6, 1, 16]	5320000.00
x[4, 6, 3]	23000000.00	x[5, 1, 16]	5320000.00	x[6, 1, 17]	216095.60
x[4, 6, 6]	167000000.00	x[5, 1, 17]	12400000.00	x[6, 1, 19]	349524.50
x[4, 6, 8]	29900000.00	x[5, 1, 19]	349524.50	x[6, 1, 24]	1680000.00
x[4, 6, 12]	22700000.00	x[5, 1, 24]	1680000.00	x[6, 1, 27]	2040000.00
x[4, 6, 13]	12600000.00	x[5, 1, 27]	2040000.00	x[6, 1, 30]	2490000.00
x[4, 6, 16]	42800000.00	x[5, 1, 30]	1550000.00	x[6, 1, 36]	1930000.00
x[4, 6, 27]	5680000.00	x[5, 1, 36]	1290000.00	x[6, 1, 46]	4160000.00
x[4, 6, 32]	2390000.00	x[5, 1, 46]	4160000.00	x[6, 1, 53]	633181.70
x[4, 6, 33]	20200000.00	x[5, 1, 52]	938541.80	x[6, 2, 40]	2060000.00
x[4, 6, 34]	60000000.00	x[5, 1, 53]	633181.70	x[6, 2, 48]	760625.40
x[4, 6, 38]	740155.10	x[5, 2, 40]	2060000.00	x[6, 5, 45]	1130000.00
x[4, 6, 46]	11000000.00	x[5, 2, 48]	760625.40	x[6, 6, 33]	2180000.00
x[4, 6, 53]	27800000.00	x[5, 5, 45]	1130000.00	x[6, 6, 50]	3200000.00
x[4, 7, 4]	4470000.00	x[5, 6, 33]	1980000.00	x[6, 7, 6]	4140000.00
x[4, 7, 18]	27400000.00	x[5, 6, 50]	3200000.00	x[6, 7, 18]	4960000.00
x[4, 7, 26]	15800000.00	x[5, 7, 6]	4050000.00	x[6, 7, 20]	1020000.00
x[4, 7, 30]	55400000.00	x[5, 7, 18]	3880000.00	x[6, 7, 26]	1080000.00
x[4, 7, 52]	16300000.00	x[5, 7, 20]	2090000.00	x[6, 7, 34]	241854.80
x[4, 8, 28]	25600000.00	x[5, 7, 26]	1080000.00	x[6, 8, 28]	1280000.00
x[4, 9, 31]	29300000.00	x[5, 7, 34]	532985.30	x[6, 9, 11]	1790000.00
x[4, 9, 49]	31300000.00	x[5, 8, 28]	1280000.00	x[6, 9, 31]	3640000.00
x[4, 10, 2]	28400000.00	x[5, 9, 10]	561372.40	x[6, 9, 49]	729444.90
x[4, 10, 14]	32900000.00	x[5, 9, 11]	2400000.00	x[6, 10, 2]	4220000.00
x[4, 10, 22]	5360000.00	x[5, 9, 31]	3040000.00	x[6, 10, 14]	3600000.00
x[4, 10, 42]	1020000.00	x[5, 9, 49]	729444.90	x[6, 10, 42]	561372.40
x[4, 10, 44]	4590000.00	x[5, 10, 2]	4220000.00	x[6, 11, 35]	1040000.00
x[4, 11, 35]	10400000.00	x[5, 10, 14]	2510000.00	x[6, 11, 37]	941355.70
x[4, 11, 47]	15700000.00	x[5, 11, 35]	1040000.00	x[6, 11, 47]	623856.90
x[4, 11, 48]	8380000.00	x[5, 11, 37]	941355.70	x[6, 12, 39]	1960000.00
x[4, 12, 39]	19600000.00	x[5, 11, 47]	623856.90	x[6, 13, 5]	2690000.00

Continued on next page

Variable	Value	Variable	Value	Variable	Value
x[4, 13, 43]	24300000.00	x[5, 12, 39]	1960000.00	x[6, 13, 23]	1450000.00
x[4, 14, 41]	10000000.00	x[5, 13, 5]	2690000.00	x[6, 13, 43]	854299.30
x[4, 15, 21]	9660000.00	x[5, 13, 23]	1450000.00	x[6, 14, 29]	1200000.00
x[4, 15, 51]	51300000.00	x[5, 13, 43]	854299.30	x[6, 14, 41]	3850000.00
x[4, 16, 25]	4970000.00	x[5, 14, 29]	1200000.00	x[6, 15, 21]	1670000.00
		x[5, 14, 41]	3850000.00	x[6, 15, 51]	714170.80
		x[5, 15, 21]	1670000.00	x[6, 16, 25]	1040000.00
		x[5, 15, 51]	1800000.00		
		x[5, 16, 25]	1040000.00		

Variable	Value	Variable	Value	Variable	Value
x[7, 1, 3]	3440000.00	x[7, 1, 46]	4160000.00	x[7, 9, 31]	2930000.00
x[7, 1, 8]	798494.10	x[7, 1, 53]	633181.70	x[7, 10, 2]	4780000.00
x[7, 1, 12]	4740000.00	x[7, 2, 40]	5180000.00	x[7, 10, 14]	3600000.00
x[7, 1, 13]	1190000.00	x[7, 2, 48]	1380000.00	x[7, 11, 35]	1990000.00
x[7, 1, 15]	629217.00	x[7, 5, 45]	1130000.00	x[7, 12, 39]	790992.70
x[7, 1, 16]	5320000.00	x[7, 6, 33]	2250000.00	x[7, 13, 5]	2690000.00
x[7, 1, 17]	216095.60	x[7, 6, 50]	2930000.00	x[7, 13, 23]	1450000.00
x[7, 1, 19]	349524.50	x[7, 7, 6]	4580000.00	x[7, 13, 43]	854299.30
x[7, 1, 24]	1680000.00	x[7, 7, 18]	3880000.00	x[7, 14, 41]	5040000.00
x[7, 1, 27]	2040000.00	x[7, 7, 20]	446966.10	x[7, 15, 21]	1670000.00
x[7, 1, 30]	2490000.00	x[7, 7, 26]	1080000.00	x[7, 15, 51]	714170.80
x[7, 1, 36]	16800000.00	x[7, 8, 28]	2560000.00	x[7, 16, 25]	1040000.00

Variable	Value	Variable	Value	Variable	Value
y[1, 1, 1]	12900000.00	y[2, 1, 5]	4570000.00	y[3, 1, 5]	3270000.00
y[1, 2, 7]	33800000.00	y[2, 2, 7]	33800000.00	y[3, 3, 1]	53900000.00
y[1, 2, 62]	8390000.00	y[2, 2, 62]	8390000.00	y[3, 3, 9]	19200000.00
y[1, 3, 9]	27400000.00	y[2, 3, 9]	27400000.00	y[3, 3, 19]	2450000.00
y[1, 3, 19]	3500000.00	y[2, 3, 19]	3500000.00	y[3, 3, 32]	1660000.00
y[1, 4, 10]	2700000.00	y[2, 3, 32]	2370000.00	y[3, 3, 35]	1510000.00
y[1, 4, 32]	2370000.00	y[2, 3, 35]	2160000.00	y[3, 3, 79]	793130.40
y[1, 4, 35]	2160000.00	y[2, 3, 79]	1130000.00	y[3, 6, 12]	11800000.00
y[1, 5, 11]	26900000.00	y[2, 4, 10]	4470000.00	y[3, 6, 18]	2540000.00
y[1, 8, 16]	5220000.00	y[2, 5, 11]	9460000.00	y[3, 6, 58]	1130000.00
y[1, 8, 26]	2760000.00	y[2, 6, 12]	16800000.00	y[3, 6, 63]	1020000.00
y[1, 12, 13]	3910000.00	y[2, 6, 18]	3630000.00	y[3, 6, 65]	5450000.00
y[1, 12, 20]	20800000.00	y[2, 6, 58]	429398.00	y[3, 6, 67]	179045.50
y[1, 12, 23]	22700000.00	y[2, 6, 65]	7780000.00	y[3, 6, 69]	5490000.00
y[1, 13, 45]	11900000.00	y[2, 6, 69]	7850000.00	y[3, 6, 94]	670322.50
y[1, 14, 25]	20800000.00	y[2, 7, 1]	8310000.00	y[3, 6, 97]	656286.80
y[1, 14, 43]	15200000.00	y[2, 8, 16]	5220000.00	y[3, 8, 26]	1940000.00
y[1, 15, 15]	3840000.00	y[2, 8, 26]	2760000.00	y[3, 12, 1]	43100000.00
y[1, 15, 31]	2450000.00	y[2, 10, 22]	4180000.00	y[3, 12, 15]	2690000.00
y[1, 16, 33]	13800000.00	y[2, 12, 20]	20800000.00	y[3, 12, 16]	3650000.00

Continued on next page

Variable	Value	Variable	Value	Variable	Value
y[1, 16, 34]	13400000.00	y[2, 13, 33]	10500000.00	y[3, 12, 20]	14600000.00
y[1, 16, 50]	10700000.00	y[2, 13, 45]	11900000.00	y[3, 12, 31]	1720000.00
y[1, 16, 56]	9650000.00	y[2, 14, 25]	20800000.00	y[3, 13, 45]	8300000.00
y[1, 16, 93]	5580000.00	y[2, 14, 43]	10300000.00	y[3, 16, 13]	2740000.00
y[1, 18, 2]	12900000.00	y[2, 15, 1]	29700000.00	y[3, 16, 23]	15900000.00
y[1, 18, 28]	2540000.00	y[2, 15, 15]	3840000.00	y[3, 16, 34]	9370000.00
y[1, 18, 36]	13600000.00	y[2, 15, 31]	2450000.00	y[3, 16, 50]	7520000.00
y[1, 18, 39]	13700000.00	y[2, 16, 13]	3910000.00	y[3, 16, 93]	3900000.00
y[1, 18, 61]	1520000.00	y[2, 16, 23]	22700000.00	y[3, 18, 2]	4570000.00
y[1, 19, 1]	35400000.00	y[2, 16, 34]	13400000.00	y[3, 19, 38]	1480000.00
y[1, 19, 17]	3640000.00	y[2, 16, 50]	10700000.00	y[3, 19, 67]	785275.70
y[1, 20, 10]	1770000.00	y[2, 16, 93]	5580000.00	y[3, 24, 33]	9670000.00
y[1, 20, 79]	1130000.00	y[2, 18, 2]	16500000.00	y[3, 26, 2]	6960000.00
y[1, 21, 41]	14400000.00	y[2, 18, 28]	2540000.00	y[3, 26, 8]	3460000.00
y[1, 21, 71]	1210000.00	y[2, 18, 30]	2470000.00	y[3, 26, 10]	3130000.00
y[1, 21, 84]	1060000.00	y[2, 18, 36]	13600000.00	y[3, 26, 28]	1780000.00
y[1, 23, 44]	7080000.00	y[2, 18, 39]	13700000.00	y[3, 26, 30]	1730000.00
y[1, 24, 24]	7490000.00	y[2, 18, 61]	1520000.00	y[3, 26, 36]	9540000.00
y[1, 25, 51]	9290000.00	y[2, 19, 5]	4670000.00	y[3, 26, 39]	9620000.00
y[1, 25, 81]	1090000.00	y[2, 19, 38]	2120000.00	y[3, 26, 52]	7590000.00
y[1, 26, 8]	4940000.00	y[2, 19, 67]	1380000.00	y[3, 26, 61]	1060000.00
y[1, 26, 52]	10800000.00	y[2, 21, 41]	14400000.00	y[3, 27, 29]	6820000.00
y[1, 27, 29]	9740000.00	y[2, 22, 43]	4940000.00	y[3, 27, 53]	7480000.00
y[1, 27, 53]	10700000.00	y[2, 24, 24]	16100000.00	y[3, 28, 3]	8180000.00
y[1, 28, 3]	11700000.00	y[2, 24, 33]	3280000.00	y[3, 28, 22]	16800000.00
y[1, 28, 22]	18700000.00	y[2, 25, 51]	9290000.00	y[3, 28, 54]	8970000.00
y[1, 28, 54]	12800000.00	y[2, 26, 8]	4940000.00	y[3, 28, 80]	779885.40
y[1, 28, 80]	1110000.00	y[2, 26, 52]	10800000.00	y[3, 28, 83]	4620000.00
y[1, 29, 55]	7200000.00	y[2, 27, 29]	9740000.00	y[3, 29, 55]	8370000.00
y[1, 31, 21]	22700000.00	y[2, 27, 53]	10700000.00	y[3, 30, 24]	11800000.00
y[1, 31, 22]	5280000.00	y[2, 28, 54]	12800000.00	y[3, 30, 40]	8060000.00
y[1, 31, 83]	6590000.00	y[2, 29, 55]	12000000.00	y[3, 30, 59]	6570000.00
y[1, 32, 1]	45100000.00	y[2, 30, 24]	779789.50	y[3, 30, 60]	7270000.00
y[1, 32, 63]	1450000.00	y[2, 30, 59]	9390000.00	y[3, 30, 70]	5590000.00
y[1, 32, 94]	957603.60	y[2, 30, 60]	10400000.00	y[3, 30, 82]	4910000.00
y[1, 32, 97]	937552.50	y[2, 30, 70]	7990000.00	y[3, 30, 99]	3600000.00
y[1, 33, 27]	2580000.00	y[2, 30, 99]	5140000.00	y[3, 31, 21]	1280000.00
y[1, 33, 46]	1970000.00	y[2, 31, 21]	22700000.00	y[3, 33, 27]	1800000.00
y[1, 33, 57]	1630000.00	y[2, 31, 22]	17200000.00	y[3, 33, 46]	1380000.00
y[1, 33, 64]	10800000.00	y[2, 31, 80]	1110000.00	y[3, 33, 57]	873794.40
y[1, 33, 90]	3870000.00	y[2, 31, 83]	6590000.00	y[3, 33, 64]	7550000.00
y[1, 33, 91]	965005.20	y[2, 33, 46]	1970000.00	y[3, 33, 90]	2710000.00
y[1, 33, 96]	945805.60	y[2, 33, 64]	10800000.00	y[3, 33, 91]	675503.60
y[1, 34, 1]	45300000.00	y[2, 33, 90]	3870000.00	y[3, 35, 66]	4050000.00
y[1, 34, 2]	3610000.00	y[2, 33, 91]	965005.20	y[3, 36, 5]	3200000.00
y[1, 34, 5]	9250000.00	y[2, 34, 58]	1180000.00	y[3, 36, 17]	2550000.00
y[1, 34, 12]	16800000.00	y[2, 34, 63]	1450000.00	y[3, 39, 51]	6500000.00
y[1, 34, 18]	3630000.00	y[2, 34, 97]	937552.50	y[3, 39, 73]	5540000.00

Continued on next page

Variable	Value	Variable	Value	Variable	Value
y[1, 34, 30]	2470000.00	y[2, 35, 66]	10400000.00	y[3, 40, 21]	14600000.00
y[1, 34, 38]	2120000.00	y[2, 36, 1]	101000000.00	y[3, 40, 48]	8900000.00
y[1, 34, 65]	7780000.00	y[2, 36, 17]	3640000.00	y[3, 40, 74]	5510000.00
y[1, 34, 67]	1380000.00	y[2, 37, 68]	4170000.00	y[3, 40, 88]	5110000.00
y[1, 34, 69]	7850000.00	y[2, 39, 3]	11700000.00	y[3, 41, 4]	17900000.00
y[1, 35, 66]	10400000.00	y[2, 39, 73]	7910000.00	y[3, 41, 14]	2720000.00
y[1, 35, 68]	9410000.00	y[2, 40, 48]	1600000.00	y[3, 41, 75]	4890000.00
y[1, 38, 58]	1610000.00	y[2, 40, 74]	7870000.00	y[3, 41, 81]	761186.70
y[1, 39, 73]	7910000.00	y[2, 41, 4]	25600000.00	y[3, 41, 85]	733832.00
y[1, 40, 24]	9350000.00	y[2, 41, 14]	3880000.00	y[3, 41, 92]	672394.90
y[1, 40, 40]	11900000.00	y[2, 41, 75]	6990000.00	y[3, 43, 44]	10200000.00
y[1, 40, 47]	14700000.00	y[2, 41, 81]	1090000.00	y[3, 43, 77]	5980000.00
y[1, 40, 48]	12700000.00	y[2, 41, 85]	1050000.00	y[3, 45, 6]	7250000.00
y[1, 40, 59]	9390000.00	y[2, 41, 92]	960564.20	y[3, 45, 98]	647965.80
y[1, 40, 60]	10400000.00	y[2, 42, 76]	5610000.00	y[3, 46, 56]	6750000.00
y[1, 40, 70]	7990000.00	y[2, 43, 11]	5110000.00	y[3, 47, 66]	3270000.00
y[1, 40, 74]	7870000.00	y[2, 43, 44]	14500000.00	y[3, 47, 86]	4370000.00
y[1, 40, 82]	7010000.00	y[2, 43, 71]	1210000.00	y[3, 48, 7]	16300000.00
y[1, 40, 88]	7290000.00	y[2, 43, 77]	8540000.00	y[3, 48, 68]	6590000.00
y[1, 40, 99]	5140000.00	y[2, 45, 6]	10400000.00	y[3, 48, 76]	3930000.00
y[1, 41, 4]	25600000.00	y[2, 45, 98]	925665.50	y[3, 48, 87]	5320000.00
y[1, 41, 14]	3880000.00	y[2, 46, 56]	9650000.00	y[3, 50, 37]	1870000.00
y[1, 41, 55]	4760000.00	y[2, 46, 82]	7010000.00	y[3, 50, 42]	10700000.00
y[1, 41, 75]	6990000.00	y[2, 47, 68]	5240000.00	y[3, 50, 49]	1280000.00
y[1, 41, 85]	1050000.00	y[2, 47, 86]	6240000.00	y[3, 50, 57]	268558.30
y[1, 41, 92]	960564.20	y[2, 48, 48]	11100000.00	y[3, 50, 78]	5000000.00
y[1, 42, 76]	5610000.00	y[2, 48, 87]	7610000.00	y[3, 50, 89]	4820000.00
y[1, 43, 44]	7440000.00	y[2, 49, 22]	2540000.00	y[3, 50, 96]	662063.90
y[1, 43, 77]	8540000.00	y[2, 49, 88]	7290000.00	y[3, 51, 7]	7300000.00
y[1, 45, 6]	10400000.00	y[2, 50, 27]	2580000.00	y[3, 51, 11]	18800000.00
y[1, 45, 98]	925665.50	y[2, 50, 37]	2670000.00	y[3, 51, 25]	14500000.00
y[1, 47, 86]	986822.70	y[2, 50, 42]	15300000.00	y[3, 51, 41]	10100000.00
y[1, 48, 86]	5250000.00	y[2, 50, 49]	1820000.00	y[3, 51, 43]	10600000.00
y[1, 48, 87]	7610000.00	y[2, 50, 57]	1630000.00	y[3, 51, 62]	5870000.00
y[1, 50, 37]	2670000.00	y[2, 50, 78]	7140000.00	y[3, 51, 71]	843570.20
y[1, 50, 42]	15300000.00	y[2, 50, 89]	6890000.00	y[3, 51, 84]	742932.00
y[1, 50, 49]	1820000.00	y[2, 50, 94]	957603.60	y[3, 51, 95]	5000000.00
y[1, 50, 78]	7140000.00	y[2, 50, 96]	945805.60	y[3, 52, 40]	274017.80
y[1, 50, 89]	6890000.00	y[2, 51, 11]	12300000.00	y[3, 52, 47]	10300000.00
y[1, 51, 95]	7140000.00	y[2, 51, 84]	1060000.00	y[3, 53, 72]	820029.30
y[1, 53, 72]	1170000.00	y[2, 51, 95]	7140000.00	y[3, 53, 100]	3610000.00
y[1, 53, 100]	5160000.00	y[2, 52, 40]	11900000.00		
		y[2, 52, 47]	14700000.00		
		y[2, 53, 72]	1170000.00		
		y[2, 53, 100]	5160000.00		

Variable	Value	Variable	Value	Variable	Value
$y[4, 2, 7]$	28400000.00	$y[5, 2, 7]$	3380000.00	$y[6, 2, 7]$	3380000.00
$y[4, 3, 9]$	27400000.00	$y[5, 2, 62]$	838614.70	$y[6, 2, 62]$	838614.70
$y[4, 4, 10]$	4470000.00	$y[5, 3, 9]$	2740000.00	$y[6, 3, 9]$	2740000.00
$y[4, 6, 1]$	139000000.00	$y[5, 3, 19]$	350387.60	$y[6, 3, 19]$	350387.60
$y[4, 6, 5]$	9250000.00	$y[5, 3, 32]$	236938.60	$y[6, 3, 32]$	236938.60
$y[4, 6, 17]$	3640000.00	$y[5, 3, 79]$	113304.30	$y[6, 3, 79]$	113304.30
$y[4, 6, 19]$	3500000.00	$y[5, 5, 11]$	2690000.00	$y[6, 5, 11]$	2690000.00
$y[4, 6, 31]$	2450000.00	$y[5, 6, 12]$	1680000.00	$y[6, 6, 12]$	1680000.00
$y[4, 6, 32]$	2370000.00	$y[5, 6, 18]$	362740.50	$y[6, 6, 18]$	362740.50
$y[4, 6, 35]$	2160000.00	$y[5, 6, 49]$	182159.70	$y[6, 6, 49]$	182159.70
$y[4, 6, 38]$	2120000.00	$y[5, 6, 58]$	161287.70	$y[6, 6, 58]$	161287.70
$y[4, 6, 67]$	1380000.00	$y[5, 6, 65]$	778319.10	$y[6, 6, 65]$	778319.10
$y[4, 6, 79]$	1130000.00	$y[5, 6, 69]$	784597.80	$y[6, 6, 69]$	784597.80
$y[4, 8, 15]$	3840000.00	$y[5, 6, 94]$	95760.36	$y[6, 6, 94]$	95760.36
$y[4, 8, 16]$	5220000.00	$y[5, 7, 15]$	383987.50	$y[6, 6, 97]$	93755.25
$y[4, 8, 20]$	20800000.00	$y[5, 8, 16]$	522023.20	$y[6, 8, 16]$	522023.20
$y[4, 12, 23]$	22700000.00	$y[5, 8, 26]$	276470.90	$y[6, 8, 26]$	276470.90
$y[4, 13, 33]$	738151.50	$y[5, 9, 1]$	1640000.00	$y[6, 9, 1]$	13200000.00
$y[4, 13, 45]$	11900000.00	$y[5, 10, 76]$	561372.40	$y[6, 11, 22]$	1790000.00
$y[4, 14, 25]$	17700000.00	$y[5, 11, 22]$	2400000.00	$y[6, 12, 13]$	391305.00
$y[4, 14, 43]$	15200000.00	$y[5, 12, 13]$	391305.00	$y[6, 12, 20]$	2080000.00
$y[4, 16, 33]$	13100000.00	$y[5, 12, 20]$	2080000.00	$y[6, 12, 23]$	2270000.00
$y[4, 16, 34]$	13400000.00	$y[5, 12, 23]$	2270000.00	$y[6, 13, 45]$	1190000.00
$y[4, 16, 50]$	10700000.00	$y[5, 13, 45]$	1190000.00	$y[6, 14, 25]$	2080000.00
$y[4, 16, 93]$	5580000.00	$y[5, 14, 25]$	988324.70	$y[6, 14, 43]$	1520000.00
$y[4, 18, 36]$	13600000.00	$y[5, 14, 43]$	1520000.00	$y[6, 15, 15]$	383987.50
$y[4, 18, 39]$	13700000.00	$y[5, 15, 31]$	245229.50	$y[6, 15, 31]$	245229.50
$y[4, 21, 41]$	9660000.00	$y[5, 16, 33]$	1380000.00	$y[6, 16, 33]$	1380000.00
$y[4, 22, 7]$	5360000.00	$y[5, 16, 34]$	1340000.00	$y[6, 16, 34]$	1340000.00
$y[4, 25, 51]$	4970000.00	$y[5, 16, 50]$	1070000.00	$y[6, 16, 50]$	1070000.00
$y[4, 26, 8]$	4940000.00	$y[5, 16, 56]$	964559.00	$y[6, 16, 56]$	964559.00
$y[4, 26, 52]$	10800000.00	$y[5, 16, 93]$	557853.90	$y[6, 16, 93]$	557853.90
$y[4, 27, 26]$	2760000.00	$y[5, 17, 1]$	12200000.00	$y[6, 17, 35]$	216095.60
$y[4, 27, 53]$	2910000.00	$y[5, 17, 35]$	216095.60	$y[6, 18, 2]$	1080000.00
$y[4, 28, 3]$	11700000.00	$y[5, 18, 8]$	493846.50	$y[6, 18, 8]$	493846.50
$y[4, 28, 54]$	12800000.00	$y[5, 18, 28]$	253801.40	$y[6, 18, 28]$	253801.40
$y[4, 28, 80]$	1110000.00	$y[5, 18, 30]$	247064.90	$y[6, 18, 30]$	247064.90
$y[4, 29, 55]$	12000000.00	$y[5, 18, 36]$	1360000.00	$y[6, 18, 36]$	1360000.00
$y[4, 30, 24]$	16800000.00	$y[5, 18, 39]$	1370000.00	$y[6, 18, 39]$	1370000.00
$y[4, 30, 59]$	9390000.00	$y[5, 18, 61]$	152097.50	$y[6, 18, 61]$	152097.50
$y[4, 30, 60]$	10400000.00	$y[5, 19, 38]$	211764.30	$y[6, 19, 38]$	211764.30
$y[4, 30, 70]$	7990000.00	$y[5, 19, 67]$	137760.20	$y[6, 19, 67]$	137760.20
$y[4, 30, 82]$	5690000.00	$y[5, 20, 2]$	1650000.00	$y[6, 20, 2]$	570416.00
$y[4, 30, 99]$	5140000.00	$y[5, 20, 10]$	446966.10	$y[6, 20, 10]$	446966.10
$y[4, 31, 21]$	22700000.00	$y[5, 21, 41]$	1440000.00	$y[6, 21, 41]$	1440000.00
$y[4, 31, 83]$	6590000.00	$y[5, 21, 71]$	120510.00	$y[6, 21, 71]$	120510.00
$y[4, 32, 63]$	1450000.00	$y[5, 21, 84]$	106133.10	$y[6, 21, 84]$	106133.10
$y[4, 32, 97]$	937552.50	$y[5, 23, 44]$	1450000.00	$y[6, 23, 44]$	1450000.00

Continued on next page

Variable	Value	Variable	Value	Variable	Value
y[4, 33, 27]	2580000.00	y[5, 24, 24]	1680000.00	y[6, 24, 24]	1680000.00
y[4, 33, 46]	1970000.00	y[5, 25, 51]	928609.10	y[6, 25, 51]	928609.10
y[4, 33, 64]	10800000.00	y[5, 25, 81]	108741.00	y[6, 25, 81]	108741.00
y[4, 33, 90]	3870000.00	y[5, 26, 52]	1080000.00	y[6, 26, 52]	1080000.00
y[4, 33, 91]	965005.20	y[5, 27, 29]	973645.40	y[6, 27, 29]	973645.40
y[4, 34, 2]	16500000.00	y[5, 27, 53]	1070000.00	y[6, 27, 53]	1070000.00
y[4, 34, 12]	16800000.00	y[5, 28, 54]	1280000.00	y[6, 28, 54]	1280000.00
y[4, 34, 18]	3630000.00	y[5, 29, 55]	1200000.00	y[6, 29, 55]	1200000.00
y[4, 34, 28]	2540000.00	y[5, 30, 60]	1040000.00	y[6, 30, 59]	938541.80
y[4, 34, 30]	2470000.00	y[5, 30, 99]	514388.00	y[6, 30, 60]	1040000.00
y[4, 34, 58]	872722.40	y[5, 31, 21]	2270000.00	y[6, 30, 99]	514388.00
y[4, 34, 61]	1520000.00	y[5, 31, 80]	111412.20	y[6, 31, 21]	2270000.00
y[4, 34, 65]	7780000.00	y[5, 31, 83]	659395.60	y[6, 31, 22]	604470.30
y[4, 34, 69]	7850000.00	y[5, 33, 27]	257624.10	y[6, 31, 80]	111412.20
y[4, 35, 66]	10400000.00	y[5, 33, 57]	163193.20	y[6, 31, 83]	659395.60
y[4, 38, 58]	740155.10	y[5, 33, 64]	1080000.00	y[6, 33, 27]	257624.10
y[4, 39, 51]	4310000.00	y[5, 33, 90]	387127.80	y[6, 33, 46]	197375.20
y[4, 39, 73]	7910000.00	y[5, 33, 96]	94580.56	y[6, 33, 57]	163193.20
y[4, 40, 48]	11900000.00	y[5, 34, 46]	197375.20	y[6, 33, 64]	1080000.00
y[4, 40, 74]	7870000.00	y[5, 34, 63]	145354.30	y[6, 33, 90]	387127.80
y[4, 41, 4]	25600000.00	y[5, 34, 91]	96500.52	y[6, 33, 96]	94580.56
y[4, 41, 14]	3880000.00	y[5, 34, 97]	93755.25	y[6, 34, 63]	145354.30
y[4, 41, 75]	6990000.00	y[5, 35, 66]	1040000.00	y[6, 34, 91]	96500.52
y[4, 41, 81]	1090000.00	y[5, 36, 5]	924688.20	y[6, 35, 66]	1040000.00
y[4, 41, 85]	1050000.00	y[5, 36, 17]	364080.00	y[6, 36, 1]	636488.40
y[4, 41, 92]	960564.20	y[5, 37, 68]	941355.70	y[6, 36, 5]	924688.20
y[4, 42, 76]	1020000.00	y[5, 39, 3]	1170000.00	y[6, 36, 17]	364080.00
y[4, 43, 44]	14500000.00	y[5, 39, 73]	790992.70	y[6, 37, 68]	941355.70
y[4, 43, 71]	1210000.00	y[5, 40, 48]	1270000.00	y[6, 39, 3]	1170000.00
y[4, 43, 77]	8540000.00	y[5, 40, 74]	787232.50	y[6, 39, 73]	790992.70
y[4, 44, 76]	4590000.00	y[5, 41, 4]	2560000.00	y[6, 40, 48]	1270000.00
y[4, 45, 6]	10400000.00	y[5, 41, 14]	388062.80	y[6, 40, 74]	787232.50
y[4, 45, 98]	925665.50	y[5, 41, 75]	699059.70	y[6, 41, 4]	2560000.00
y[4, 46, 56]	9650000.00	y[5, 41, 85]	104833.10	y[6, 41, 14]	388062.80
y[4, 46, 82]	1320000.00	y[5, 41, 92]	96056.42	y[6, 41, 75]	699059.70
y[4, 47, 68]	9410000.00	y[5, 43, 77]	854299.30	y[6, 41, 85]	104833.10
y[4, 47, 86]	6240000.00	y[5, 45, 6]	1040000.00	y[6, 41, 92]	96056.42
y[4, 48, 48]	778208.20	y[5, 45, 98]	92566.55	y[6, 42, 76]	561372.40
y[4, 48, 87]	7610000.00	y[5, 46, 40]	1190000.00	y[6, 43, 77]	854299.30
y[4, 49, 22]	24000000.00	y[5, 46, 47]	1470000.00	y[6, 45, 6]	1040000.00
y[4, 49, 88]	7290000.00	y[5, 46, 70]	798882.20	y[6, 45, 98]	92566.55
y[4, 50, 37]	2670000.00	y[5, 46, 82]	700751.20	y[6, 46, 40]	1190000.00
y[4, 50, 42]	15300000.00	y[5, 47, 86]	623856.90	y[6, 46, 47]	1470000.00
y[4, 50, 49]	1820000.00	y[5, 48, 87]	760625.40	y[6, 46, 70]	798882.20
y[4, 50, 57]	1630000.00	y[5, 49, 88]	729444.90	y[6, 46, 82]	700751.20
y[4, 50, 78]	7140000.00	y[5, 50, 37]	267012.20	y[6, 47, 86]	623856.90
y[4, 50, 89]	6890000.00	y[5, 50, 42]	1530000.00	y[6, 48, 87]	760625.40
y[4, 50, 94]	957603.60	y[5, 50, 78]	714295.30	y[6, 49, 88]	729444.90

Continued on next page

Variable	Value	Variable	Value	Variable	Value
y[4, 50, 96]	945805.60	y[5, 50, 89]	688941.60	y[6, 50, 37]	267012.20
y[4, 51, 11]	26900000.00	y[5, 51, 25]	1090000.00	y[6, 50, 42]	1530000.00
y[4, 51, 25]	3030000.00	y[5, 51, 95]	714170.80	y[6, 50, 78]	714295.30
y[4, 51, 41]	4790000.00	y[5, 52, 59]	938541.80	y[6, 50, 89]	688941.60
y[4, 51, 62]	8390000.00	y[5, 53, 72]	117147.00	y[6, 51, 95]	714170.80
y[4, 51, 84]	1060000.00	y[5, 53, 100]	516034.70	y[6, 53, 72]	117147.00
y[4, 51, 95]	7140000.00			y[6, 53, 100]	516034.70
y[4, 52, 40]	11900000.00				
y[4, 52, 47]	14700000.00				
y[4, 53, 13]	3910000.00				
y[4, 53, 29]	9740000.00				
y[4, 53, 53]	7780000.00				
y[4, 53, 72]	1170000.00				
y[4, 53, 100]	5160000.00				

Variable	Value	Variable	Value	Variable	Value
y[7, 2, 7]	3380000.00	y[7, 17, 35]	216095.60	y[7, 33, 96]	94580.56
y[7, 2, 62]	838614.70	y[7, 18, 8]	493846.50	y[7, 35, 66]	1040000.00
y[7, 2, 76]	561372.40	y[7, 18, 28]	253801.40	y[7, 35, 68]	941355.70
y[7, 3, 9]	2740000.00	y[7, 18, 30]	247064.90	y[7, 36, 1]	13900000.00
y[7, 3, 19]	350387.60	y[7, 18, 36]	1360000.00	y[7, 36, 2]	1650000.00
y[7, 3, 32]	236938.60	y[7, 18, 39]	1370000.00	y[7, 36, 5]	924688.20
y[7, 3, 79]	113304.30	y[7, 18, 61]	152097.50	y[7, 36, 17]	364080.00
y[7, 5, 11]	2690000.00	y[7, 19, 38]	211764.30	y[7, 39, 73]	790992.70
y[7, 6, 12]	1680000.00	y[7, 19, 67]	137760.20	y[7, 40, 22]	2400000.00
y[7, 6, 18]	362740.50	y[7, 20, 10]	446966.10	y[7, 40, 48]	1270000.00
y[7, 6, 46]	197375.20	y[7, 21, 41]	1440000.00	y[7, 40, 74]	787232.50
y[7, 6, 49]	182159.70	y[7, 21, 71]	120510.00	y[7, 40, 88]	729444.90
y[7, 6, 58]	161287.70	y[7, 21, 84]	106133.10	y[7, 41, 4]	2560000.00
y[7, 6, 63]	145354.30	y[7, 23, 44]	1450000.00	y[7, 41, 14]	388062.80
y[7, 6, 65]	778319.10	y[7, 24, 24]	1680000.00	y[7, 41, 55]	1200000.00
y[7, 6, 69]	784597.80	y[7, 25, 51]	928609.10	y[7, 41, 75]	699059.70
y[7, 6, 91]	96500.52	y[7, 25, 81]	108741.00	y[7, 41, 85]	104833.10
y[7, 6, 94]	95760.36	y[7, 26, 52]	1080000.00	y[7, 41, 92]	96056.42
y[7, 6, 97]	93755.25	y[7, 27, 29]	973645.40	y[7, 43, 77]	854299.30
y[7, 8, 16]	522023.20	y[7, 27, 53]	1070000.00	y[7, 45, 6]	1040000.00
y[7, 8, 26]	276470.90	y[7, 28, 3]	1170000.00	y[7, 45, 98]	92566.55
y[7, 12, 13]	391305.00	y[7, 28, 54]	1280000.00	y[7, 46, 40]	1190000.00
y[7, 12, 20]	2080000.00	y[7, 28, 80]	111412.20	y[7, 46, 47]	1470000.00
y[7, 12, 23]	2270000.00	y[7, 30, 59]	938541.80	y[7, 46, 70]	798882.20
y[7, 13, 45]	1190000.00	y[7, 30, 60]	1040000.00	y[7, 46, 82]	700751.20
y[7, 14, 25]	2080000.00	y[7, 30, 99]	514388.00	y[7, 48, 86]	623856.90
y[7, 14, 43]	1520000.00	y[7, 31, 21]	2270000.00	y[7, 48, 87]	760625.40
y[7, 15, 15]	383987.50	y[7, 31, 83]	659395.60	y[7, 50, 42]	1530000.00
y[7, 15, 31]	245229.50	y[7, 33, 27]	257624.10	y[7, 50, 78]	714295.30
y[7, 16, 33]	1380000.00	y[7, 33, 37]	267012.20	y[7, 50, 89]	688941.60
y[7, 16, 34]	1340000.00	y[7, 33, 57]	163193.20	y[7, 51, 95]	714170.80

Continued on next page

Variable	Value	Variable	Value	Variable	Value
$y[7, 16, 50]$	1070000.00	$y[7, 33, 64]$	1080000.00	$y[7, 53, 72]$	117147.00
$y[7, 16, 56]$	964559.00	$y[7, 33, 90]$	387127.80	$y[7, 53, 100]$	516034.70
$y[7, 16, 93]$	557853.90				

Variable	Value	Variable	Value	Variable	Value
$z[1]$	1	$z[19]$	1	$z[37]$	1.00
$z[2]$	1	$z[20]$	1	$z[38]$	1.00
$z[3]$	1	$z[21]$	1	$z[39]$	1.00
$z[4]$	1	$z[22]$	1	$z[40]$	1.00
$z[5]$	1	$z[23]$	1	$z[41]$	1.00
$z[6]$	1	$z[24]$	1	$z[42]$	1.00
$z[7]$	1	$z[25]$	1	$z[43]$	1.00
$z[8]$	1	$z[26]$	1	$z[44]$	1.00
$z[9]$	1	$z[27]$	1	$z[45]$	1.00
$z[10]$	1	$z[28]$	1	$z[46]$	1.00
$z[11]$	1	$z[29]$	1	$z[47]$	1.00
$z[12]$	1	$z[30]$	1	$z[48]$	1.00
$z[13]$	1	$z[31]$	1	$z[49]$	1.00
$z[14]$	1	$z[32]$	1	$z[50]$	1.00
$z[15]$	1	$z[33]$	1	$z[51]$	1.00
$z[16]$	1	$z[34]$	1	$z[52]$	1.00
$z[17]$	1	$z[35]$	1	$z[53]$	1.00
$z[18]$	1	$z[36]$	1		

Appendix B - Code

The code along with its previous versions can be found in this GitHub repository: https://github.com/nathanansel28/40.002-Optimisation-1D.git.

Julia

```
using JuMP, GLPK
using DataFrames, CSV, XLSX
```

Listing 1: Setting up Julia

```
# DATA: FISH
df_fish = DataFrame(CSV.File("fish_data.csv"))
fish_price = df_fish."2021 Market Price"
fish_health_value = df_fish."Protein Value"

# DATA: FISHING GROUND
df_fishingground_max = DataFrame(CSV.File("fishingground_restriction.csv"))

# DATA: PROCESSING FACILITY
df_processingfacility = DataFrame(CSV.File("processing_facility.csv"))
processing_cap = df_processingfacility."Processing Capacity"
operational_cost = df_processingfacility."Operational Cost"

# DATA: CITIES
df_cities = DataFrame(CSV.File("cities_data.csv"))
cities_budget = df_cities."GDP per Capita"
cities_population = df_cities."Population"

# DATA FOR OBJECTIVE FUNCTION
fishing_cost_3d = [
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 1")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 2")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 3")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 4")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 5")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 6")...),
    DataFrame(XLSX.readtable("fishing_cost.xlsx", "Fish 7")...)
]
transportation_cost_3d = [
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 1")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 2")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 3")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 4")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 5")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 6")...),
    DataFrame(XLSX.readtable("transportation_cost.xlsx", "Fish 7")...)
]
```

Listing 2: Data Extraction

```
# PARAMETERS
F = 7       # number of fish types
G = 16      # number of fishing grounds
I = 53      # number of processing facilities
J = 100     # number of cities
# VARIABLES
m = Model(GLPK.Optimizer)
@variable(m, x[1:F, 1:G, 1:I])
@variable(m, y[1:F, 1:I, 1:J])
@variable(m, z[1:I], Bin)
@variable(m, s[1:J])
```

Listing 3: Declaring Decision Variables

```
# CONSTRAINTS
# Non-negativity constraints
for f in 1:F
    for g in 1:G
        for i in 1:I
            @constraint(m, x[f, g, i] >= 0)
        end
    end
end
for f in 1:F
    for i in 1:I
        for j in 1:J
            @constraint(m, y[f, i, j] >= 0)
        end
    end
end
for j in 1:J
    @constraint(m, s[j] >= 0)
end

# Fishing Restriction Constraint
for f in 1:F
    for g in 1:G
        @constraint(m, sum(x[f, g, i] for i in 1:I) <= df_fishingground_max[g, f])
    end
end

# Processing Capacity Constraint
for i in 1:I
    @constraint(m, sum(y[f, i, j] for f in 1:F for j in 1:J) <= processing_cap[i])
end

# Protein Consumption Constraint
for j in 1:J
    @constraint(m, sum(y[f, i, j]*fish_health_value[f] for f in 1:F for i in 1:I) <=
    43800 * cities_population[j])
    @constraint(m, sum(y[f, i, j]*fish_health_value[f] for f in 1:F for i in 1:I) >=
    5840 * cities_population[j])
end

# Distribution Capacity Constraint
for f in 1:F
    for i in 1:I
        @constraint(m, sum(y[f, i, j] for j in 1:J) <= sum(x[f, g, i] for g in 1:G))
    end
end

# Budget Constraint
for j in 1:J
    @constraint(m, sum(y[f, i, j] * fish_price[f] for f in 1:F for i in 1:I)  <=
    cities_population[j] * (0.15) * cities_budget[j] + s[j])
end

# Fish Diversity Constraint
for f in 1:F
    for j in 1:J
        @constraint(m, 0.025 * sum(y[f_sum_index, i, j] for f_sum_index in 1:F for i
    in 1:I) <= sum(y[f, i, j] for i in 1:I))
        @constraint(m, 0.25 * sum(y[f_sum_index, i, j] for f_sum_index in 1:F for i
    in 1:I) >= sum(y[f, i, j] for i in 1:I))
    end
end

# Fishing Ground Diversity Constraint
```

```
for g in 1:G
    @constraint(m, 0.025 * sum(x[f, g_sum_index, i] for f in 1:F for g_sum_index in
    1:G for i in 1:I) <= sum(x[f,g,i] for f in 1:F for i in 1:I))
    @constraint(m, 0.25 * sum(x[f, g_sum_index, i] for f in 1:F for g_sum_index in 1:
    G for i in 1:I) >= sum(x[f,g,i] for f in 1:F for i in 1:I))
end

# Facility Opening Constraint
# Part 1
for i in 1:I
    @constraint(m, sum(x[f,g,i] for f in 1:F for g in 1:G) <= sum(z[i] *
    df_fishingground_max[g, f] for f in 1:F for g in 1:G) )
end
# Part 2
for i in 1:I
    @constraint(m, sum(y[f,i,j] for f in 1:F for j in 1:J) <= z[i] * processing_cap[i
    ])
end
```

Listing 4: Constraints

```
# OBJECTIVE FUNCTION
@objective(m, Max, (10^5)*sum(fish_health_value[f]*y[f, i, j] for f in 1:F for i in
    1:I for j in 1:J) - sum(fishing_cost_3d[f][g, i]*x[f, g, i] for f in 1:F for g in
     1:G for i in 1:I) - sum(transportation_cost_3d[f][i,j]*y[f, i, j] for f in 1:F
    for i in 1:I for j in 1:J) - sum(operational_cost[i]*z[i] for i in 1:I) - sum(s[j
    ] for j in 1:J))
```

Listing 5: Defining the Objective Function

```
optimize!(m)
```

Listing 6: Solving the Model

```
solution_summary(m)
```

Listing 7: Printing the Solution Summary

```
objective_value(m)
```

Listing 8: Printing the Optimal Objective Value

```
println("Optimal solution for x:")
for f in 1:F
    for g in 1:G
        for i in 1:I
            println("x[$f, $g, $i] = ", value(x[f, g, i]))
        end
    end
end

println("\nOptimal solution for y:")
for f in 1:F
    for i in 1:I
        for j in 1:J
            println("y[$f, $i, $j] = ", value(y[f, i, j]))
        end
    end
end

println("\nOptimal solution for z:")
for i in 1:I
    println("z[$i] = ", value(z[i]))
end
```

```

println("\nOptimal solution for s:")
for j in 1:J
    println("s[$j] = ", value(s[j]))
end
```

Listing 9: Printing the Optimal Solution

Python

```python
from math import radians, sin, cos, sqrt, atan2
def haversine(lat1, lon1, lat2, lon2):
    # Convert latitude and longitude from degrees to radians
    lat1, lon1, lat2, lon2 = map(radians, [lat1, lon1, lat2, lon2])

    # Haversine formula
    dlat = lat2 - lat1
    dlon = lon2 - lon1
    a = sin(dlat / 2) ** 2 + cos(lat1) * cos(lat2) * sin(dlon / 2) ** 2
    c = 2 * atan2(sqrt(a), sqrt(1 - a))

    # Radius of the Earth in kilometers
    R = 6371.0

    # Calculate the distance in km
    distance = R * c
    return distance
```

Listing 10: A function to compute the distance (in km) between two points given the latitude and longitude of both points. The function uses the haversine formula.

```python
# DATA EXTRACTION
import pandas as pd
df_fishinggrounds = pd.read_csv("Model 1//Model 1 - fishing grounds.csv")
df_facility = pd.read_csv("Model 1//Model 1 - processing facilities.csv")
df_cities = pd.read_csv("Model 1//Model 1 - cities.csv")
# VISUALIZATION
display(df_fishinggrounds.head())
display(df_fishinggrounds.shape)
display(df_facility.head())
display(df_facility.shape)
display(df_cities.head())
display(df_cities.shape)
```

Listing 11: Data Extraction & Visualization

```python
import random
random.seed(42)     #set seed to 42 to ensure reproducibility of random data

# DISTANCE CALCULATION
df_distances_1 = pd.DataFrame()
for g in range(len(df_fishinggrounds)):
    lat_fg, lon_fg = df_fishinggrounds.at[g, 'Lat'], df_fishinggrounds.at[g, 'Lon']
    for i in range(len(df_facility)):
        lat_facil, lon_facil = df_facility.at[i, 'Lat'], df_facility.at[i, 'Lon']
        distance = haversine(lat_fg, lon_fg, lat_facil, lon_facil)
        df_distances_1.at[g+1, i+1] = distance
        # index adjusted to fit julia's indexing system

# RANDOMIZING: To make the dataset more interesting, we fluctuate the data randomly
    by multiplying the original distance with a random number between 0.7 and 1.3.
F = 7
f_cost_multiplier = [random.uniform(0.7, 1.3) for _ in range(10)]
dict_1 = {}
```

```
for f in range(F):
    dict_1[f] = df_distances_1 * f_cost_multiplier[f]

# EXPORT: To save the dataset in a suitable format for Julia
with pd.ExcelWriter("fishing_cost.xlsx") as writer:
    for key, df in dict_1.items():
        sheetname = f"Fish {str(1+key)}"  # Use the key as the sheet name
        df.to_excel(writer, sheet_name=sheetname, index=False)
```

Listing 12: Calculating distance between each fishing ground and each processing facility

```
# DISTANCE CALCULATION
df_distances_2 = pd.DataFrame()
for i in range(len(df_facility)):
    lat_facil, lon_facil = df_facility.at[i, 'Lat'], df_facility.at[i, 'Lon']
    for j in range(len(df_cities)):
        lat_city, lon_city = df_cities.at[j, 'Lat'], df_cities.at[j, 'Lon']
        distance = haversine(lat_facil, lon_facil, lat_city, lon_city)
        df_distances_2.at[i+1, j+1] = distance
        # index adjusted to fit julia's indexing system

# RANDOMIZING: To make the dataset more interesting, we fluctuate the data randomly
    by multiplying the original distance with a random number between 0.7 and 1.3.
F = 7
f_cost_multiplier = [random.uniform(0.7, 1.3) for _ in range(10)]
dict_2 = {}
for f in range(F):
    dict_2[f] = df_distances_2 * f_cost_multiplier[f]

# EXPORT: To save the dataset in a suitable format for Julia
with pd.ExcelWriter("transportation_cost.xlsx") as writer:
    for key, df in dict_2.items():
        sheetname = f"Fish {str(1+key)}"  # Use the key as the sheet name
        df.to_excel(writer, sheet_name=sheetname, index=False)
```

Listing 13: Calculating distance between each processing facility and each city

Appendix C - Data

Fish Data

The data can be found in the '`fish_data.csv`' file.

index f	Fish Name	2021 Market Price	Protein Value
1	Milkfish	185.350000	263.530000
2	Tilapia	148.480000	261.020000
3	Galunggong	185.280000	239.860000
4	Yellowfin Tuna	308.000000	291.760000
5	Indian Mackarel	267.000000	209.090000
6	Skipjack Tuna	181.000000	213.480000
7	Shrimp	596.490000	229.280000

Fishing Ground Data

Fishing Ground Restrictions

The data can be found in the '`fishingground_restriction.csv`' file. The column index refer to the index of the fish (1 to 7), whereas the row index refer to the index of the fishing ground (1 to 16).

1	2	3	4	5	6	7
187875742.20	836756169.40	862141729.30	14645800.48	58003552.54	452149416.50	85251294.79
553528279.10	9473784.10	385547237.80	19802445.59	544477622.60	189655109.90	180576834.00
75738737.69	34475926.38	124387583.40	41505737.82	372064473.50	539959093.90	448118063.60
552871791.00	304030593.00	826035701.80	107344872.20	41274656.78	148043647.40	777097539.40
460422592.30	328252023.50	127042609.10	750473884.70	62913522.98	400421357.10	71980113.36
363391796.20	99414934.37	35254356.95	467977870.40	384823121.10	77650155.00	262373570.20
122127000.00	326347374.00	78616565.35	119334136.70	520792903.10	606507572.10	68030800.52
562346692.70	169161088.10	146896979.40	30410853.72	348974282.90	311614496.80	663486540.60
109867567.60	406213285.50	68384559.47	237817359.60	225133283.00	129518750.60	14577500.31
112281760.40	189052283.60	18143313.91	203046221.10	6723126.34	214719072.10	134028259.40
25403148.54	807345367.40	299273007.10	428255206.90	16299847.80	538297959.60	193803622.10
174779498.60	57631178.99	646339429.40	335564302.40	471313926.80	134609097.30	198307591.60
428793711.30	853274935.90	569838121.10	536622832.90	664261339.30	303706975.80	198707899.40
450787307.80	163199095.70	542934485.30	113846119.30	152245648.40	683398495.70	655975613.70
458711211.60	321601244.80	176967689.20	545152628.80	76217882.55	56677172.33	916285203.60
214280770.20	596035224.00	86835042.55	176968285.80	228467080.80	197147296.40	353971174.20

Fishing Ground Location

The data can be found in the '`fishingground_location.csv`' file.

index g	Name of Region	Lat	Lon
1	NCR	14.599500	120.984200
2	Region V	13.139600	123.740000
3	Region XII	6.116400	125.171600
4	Region I	18.198300	120.593600
5	Region II	17.613100	121.726900
6	Region III	15.034300	120.688900
7	Region IV-A	14.676000	121.043700
8	MIMAROPA	9.967200	118.785600

Continued on next page

index g	Name of Region	Lat	Lon
9	Region VI	10.720200	122.562100
10	Region VII	10.315700	123.885400
11	Region VIII	11.244500	125.003200
12	Region IX	6.921400	122.079000
13	Region X	8.454200	124.631900
14	Region XI	7.190700	125.455300
15	Region XIII	8.949500	125.543600
16	BARMM	7.209500	124.241400

Processing Facility

The data can be found in the `'processing_facility.csv'` file.

index i	Processing Facility	Lat	Lon	Operational Cost	Processing Capacity
1	Navotas Fish	14.646386	120.949990	84619.318330	20747658.970000
2	Cebu City	10.289767	123.892048	97099.639530	125916582.800000
3	Taguig City	14.518694	121.052371	198342.170300	184836774.000000
4	Pasig City	14.549008	121.120653	49229.243100	16168077.560000
5	Cagayan de Oro	8.518487	124.639592	38051.952820	44446419.250000
6	Valenzuela	14.710667	120.936968	14591.606030	244910600.000000
7	City of Paranaque	14.503099	120.989103	8090.332418	8695362.469000
8	Bacoor	14.459890	120.932010	31756.182970	50175570.490000
9	Tondo	14.605385	120.960464	19108.310020	14871741.650000
11	Pasil	10.915890	123.038960	28235.091410	4742982.063000
12	Iloilo	10.688774	122.557353	144095.520100	4187081.432000
13	Muntinlupa City	14.404822	121.047524	30322.807790	170897271.900000
14	City of Calamba	14.280550	121.148610	84257.748790	58716340.700000
15	Lapu-Lapu City	10.311705	123.943301	57436.385220	109630664.400000
16	Pasay City	14.502569	120.995622	146781.642900	43797964.720000
17	Binan	14.332514	121.100782	95542.137150	207629372.300000
18	Sampaloc	14.602272	120.995650	93184.271010	12878744.900000
19	Taytay	14.550666	121.124033	13124.446370	139324805.500000
20	Malabon	14.653748	120.949687	17189.411090	50515947.720000
21	Cainta	14.550902	121.116103	147481.146300	6462901.339000
22	Butuan	8.957470	125.551351	91999.412210	45836896.410000
23	Mandaue City	10.352178	123.936843	205718.155700	10291124.090000
24	Iligan	8.233084	124.234590	122365.337800	11439558.410000
25	Cabuyao	14.280774	121.149341	116854.432400	41539590.300000
26	Cotabato	7.225357	124.248898	64918.340400	27745750.880000
27	Binangonan	14.462120	121.192137	106071.035400	95436344.890000
28	Tanza	14.406718	120.850855	90699.849110	66955913.850000
29	Puerto Princesa	9.726430	118.775660	70181.388080	127175705.900000
30	Tagum	7.363690	125.818790	124691.132900	41877917.760000
31	Lucena	13.903615	121.622865	47485.018690	143416153.700000
32	Talisay	10.745270	122.960600	177455.268500	122380831.800000
33	Pamarawan	14.759200	120.815480	82466.688400	50886642.110000
34	Banicain	14.828880	120.273860	43013.151510	81922929.640000
35	Marilao	14.749719	120.915496	62411.194170	164494683.900000
36	Tacloban	11.247266	124.998434	52732.977170	48885006.580000
37	Navotas	14.642660	120.951749	98109.372570	130079223.600000
38	Ormoc	11.005304	124.603195	33083.099950	6055459.193000
39	Meycauayan	14.752031	120.910914	96140.730580	2353032.629000
41	Pagadian	7.820107	123.446122	55524.176210	56476004.230000

Continued on next page

index i	Processing Facility	Lat	Lon	Operational Cost	Processing Capacity
42	Legazpi City	13.139969	123.762420	64831.225420	176388547.300000
43	Panabo	7.289824	125.695206	18608.121250	162834463.700000
44	Toledo	10.377546	123.637101	8155.785518	12811672.890000
45	Marawi City	7.984405	124.279630	213588.765200	88347319.270000
46	San Carlos City	10.477787	123.422999	116976.995900	4590871.305000
47	Santa Ana	18.509627	122.150482	60582.607880	45129325.150000
48	Tanauan	14.056993	121.077163	41103.054980	46840350.570000
49	Calbayog City	12.066563	124.594307	118073.325600	37000211.430000
50	Sorsogon	13.040950	124.045350	35649.016850	75039746.480000
51	Roxas City	11.596422	122.722769	114653.630800	42543687.770000
52	Dagupan City	16.043332	120.337221	74798.121760	145024980.400000
54	Mabua	9.805780	125.439740	21213.565760	156136876.800000
55	Sariaya	13.862587	121.506273	60704.411260	64620189.870000
56	Naic	14.320246	120.755567	198949.873900	46752024.730000

Cities Data

The data can be found in the `'cities_data.csv'` file.

index j	Name of City	Lat	Lon	Population	Status	GDP per Capita
1	Manila	14.595800	120.977200	24922000	primary	257835.470000
2	Quezon City	14.650000	121.047500	2960048	admin	238272.050000
3	Zamboanga City	6.904200	122.076100	2100000	admin	246222.580000
4	Davao	7.070000	125.600000	1776949	admin	223413.360000
5	Caloocan City	14.650000	120.970000	1661584	admin	247636.900000
6	Canagatan	18.000000	121.800000	1273219	0	201924.090000
7	Cebu City	10.320000	123.750000	964169	admin	242673.740000
8	Antipolo	14.584200	121.176300	887399	admin	216962.430000
9	Taguig City	14.520000	121.050000	886722	admin	214927.420000
10	Pasig City	14.560500	121.076500	803159	admin	233379.960000
11	Cagayan de Oro	8.480000	124.650000	728402	admin	215775.540000
12	Valenzuela	14.700000	120.980000	714978	admin	258318.960000
13	Dasmarinas	14.330000	120.940000	703141	0	185796.780000
14	General Santos	6.120000	125.170000	697315	admin	219134.850000
15	City of Paranaque	14.500800	120.991500	689992	admin	220745.810000
16	Bacoor	14.462400	120.964500	664625	0	212839.490000
17	Tondo	14.617000	120.967000	654220	0	201229.170000
18	San Jose del Monte	14.813900	121.045300	651813	0	203801.560000
19	Makati City	14.556700	121.021400	629616	admin	257316.850000
20	Las Pinas City	14.450000	120.980000	606293	0	200488.840000
21	Bacolod	10.676500	122.950900	600783	admin	220558.650000
22	Iloilo	10.720000	122.570000	574000	admin	243825.100000
23	Muntinlupa City	14.380000	121.050000	543445	admin	247300.500000
24	City of Calamba	14.220000	121.170000	539671	0	182152.390000
25	Lapu-Lapu City	10.312700	123.948800	497604	admin	245850.640000
26	Imus	14.429700	120.936700	496794	0	199039.770000
27	Angeles City	15.147200	120.584700	462928	admin	257774.070000
28	Marikina City	14.650000	121.100000	456059	admin	259280.790000
29	General Trias	14.380000	120.880000	450583	0	178312.550000
30	Rodriguez	14.720000	121.120000	443954	0	202008.980000
31	Pasay City	14.550000	121.000000	440656	admin	248084.660000
32	Mandaluyong City	14.580000	121.030000	425758	admin	221802.400000
33	Santa Rosa	14.320000	121.120000	414812	0	194491.190000

Continued on next page

index j	Name of City	Lat	Lon	Population	Status	GDP per Capita
34	Binan	14.330000	121.080000	407437	0	191924.720000
35	Sampaloc	14.603100	121.002200	388305	0	209066.780000
36	Taytay	14.569200	121.132500	386451	0	205925.810000
37	Tarlac City	15.486900	120.590000	385398	admin	220610.770000
38	Malabon	14.660000	120.960000	380522	admin	258219.460000
39	Cainta	14.570000	121.120000	376933	0	212896.750000
40	Lipa City	13.941100	121.162200	372931	0	186369.300000
41	Butuan	8.948000	125.543000	372910	admin	226287.020000
42	Baguio City	16.411900	120.593300	366358	admin	256888.050000
43	Mandaue City	10.330000	123.930000	364116	admin	258541.650000
44	Iligan	8.230000	124.250000	363115	admin	233567.110000
45	Cabuyao	14.275000	121.125000	355330	0	194933.420000
46	San Fernando	15.030000	120.680000	354666	admin	233991.170000
47	Batangas	13.830000	121.000000	351437	admin	243910.390000
48	Naga City	13.624400	123.186400	342769	admin	216514.680000
49	Cabanatuan City	15.490800	120.967800	327325	0	195612.540000
50	San Pedro	14.358300	121.058300	326001	0	192418.270000
51	Cotabato	7.220000	124.250000	325079	admin	235687.410000
52	Binangonan	14.451400	121.191900	313631	0	201766.150000
53	Tanza	14.394400	120.853100	312116	0	200001.210000
54	Puerto Princesa	9.750000	118.750000	307079	admin	260491.380000
55	Tagum	7.447800	125.807800	296202	admin	235789.280000
56	Silang	14.230600	120.975000	295644	0	215557.980000
57	Mabalacat	15.220000	120.580000	293244	0	198441.590000
58	Santa Maria	14.820000	120.960000	289820	0	186610.150000
59	San Pablo	14.070000	121.325000	285348	0	192078.710000
60	Lucena	13.930000	121.620000	278924	admin	217403.080000
61	San Mateo	14.696900	121.121900	273306	0	193464.600000
62	Talisay	10.250000	123.830000	263048	0	186177.800000
63	Malolos	14.843300	120.811400	261189	admin	250873.030000
64	Olongapo	14.830000	120.280000	260317	admin	241981.980000
65	Marilao	14.758100	120.948100	254453	0	178628.420000
66	Tacloban	11.240000	125.000000	251881	admin	242266.270000
67	Navotas	14.666700	120.941700	247543	admin	220508.900000
68	Ormoc	11.010600	124.607500	230998	admin	237983.020000
69	Meycauayan	14.730000	120.950000	225673	0	203033.600000
70	Santo Tomas	14.080000	121.180000	218500	0	213516.650000
71	Valencia	7.904200	125.092800	216546	0	210466.490000
72	Trece Martires City	14.280000	120.870000	210503	admin	249352.100000
73	Pagadian	7.827200	123.436400	210452	admin	219492.580000
74	Legazpi City	13.130000	123.730000	209533	admin	219407.290000
75	Panabo	7.300000	125.680000	209230	0	195115.040000
76	Toledo	10.380000	123.650000	207314	0	199772.200000
77	Marawi City	8.003100	124.285000	207010	admin	241001.190000
78	San Carlos City	15.928100	120.348900	205424	0	203061.240000
79	Santa Ana	14.580000	121.012000	203598	0	213670.640000
80	Kabankalan	9.980000	122.820000	200198	0	195914.600000
81	Koronadal	6.500000	124.850000	195398	0	206634.550000
82	Tanauan	14.080000	121.150000	193936	0	211011.380000
83	Bago	10.538800	122.838400	191210	0	201389.080000
84	Malaybalay	8.156400	125.133300	190712	admin	239852.200000
85	Digos	6.750000	125.350000	188376	admin	253649.560000
86	Calbayog City	12.070000	124.600000	186960	0	194866.290000
87	Sorsogon	12.974200	124.005800	182237	admin	256708.000000
88	Roxas City	11.589400	122.751100	179292	admin	237592.130000

Continued on next page

index j	Name of City	Lat	Lon	Population	Status	GDP per Capita
89	Dagupan City	16.043000	120.334000	174302	admin	230823.750000
90	Lubao	14.930000	120.600000	173502	0	189024.610000
91	Mexico	15.070000	120.720000	173403	0	216233.160000
92	Polomolok	6.220000	125.070000	172605	0	179637.240000
93	General Mariano Alvarez	14.300000	121.000000	172433	0	188929.850000
94	San Miguel	15.145800	120.978300	172073	0	180470.360000
95	Surigao	9.789700	125.495800	171107	admin	255016.500000
96	Concepcion	15.324900	120.655400	169953	0	216908.340000
97	Baliuag	14.954000	120.901000	168470	0	214663.660000
98	Tuguegarao	17.613300	121.730300	166334	admin	238750.590000
99	Sariaya	13.970000	121.530000	161868	0	185579.610000
100	Naic	14.320000	120.770000	160987	0	187192.540000

Fishing Cost

The data can be found in the `'fishing_cost.xlsx'` file. The row index refer to the fishing ground index (1 to 16), the column index refer to the processing facility index (1 to 53), and the cell contain c_{fgi}, the total cost associated with the effort of bringing 1 kg of fish f from fishing ground g to processing facility i. Each sheet contain data for 1 fish.

Unfortunately, this dataset is too big to include here.

Transportation Cost

The data can be found in the `'transportation_cost.xlsx'` file. The row index refer to the processing facility index (1 to 53), the column index refer to the city index (1 to 100), and the cell contain d_{fgi}, the total transportation cost associated with the effort of bringing 1 kg of fish f from processing facility i to city j.

Unfortunately, this dataset is too big to include here.