Bachelor of Technology Project (MCD411)

Efficient Food grain Supply Chain for Uttarakhand

Project Code. I-10

Submitted by

Karan Mittal (2017ME20671)
Utsav Khandelwal (2017ME20701)

Supervised by

Prof. Nomesh Bhojkumar Bolia

Department of Mechanical Engineering
Indian Institute of Technology Delhi
(July 2020 – January 2021)

Acknowledgements

Through the medium of this section, we would like to express our gratitude to all the people associated with us throughout this project.

Firstly, we would sincerely thank Prof Nomesh Bolia for giving us this opportunity to be a part of this project. Words might not be able to describe the amount of support he has provided to us constantly throughout. Since the time they took our course MCL262, we were determined to pursue a project under his guidance, and we are grateful to have received this learning opportunity. We hope to acquire the qualities such as politeness and determination as sir possesses.

We would also like to thank our senior PhD and MTech students as well as fellow batch mates. They were always eager to provide help of any kind, even on shortest of notice. Working together as part of team broadened our perspective and we believe helped in the overall growth of the team. We further extend our gratitude to Bolia Sir for putting such a team to work together.

Lastly, we are thankful of being part of the IITD family, who gave us all the top notch infrastructure and resources to pursue our work.

Abstract

The state of Uttarakhand is a hilly state with more than 60% area covered with forests. Difficult terrain, limited availability of transporters, restricted windows for transportation and the disintegrated storage spaces have led to the state government incurring very high costs for transportation of TPDS commodities. During monsoon and winter seasons some high-altitude areas get cut off from the main supply hubs, causing reduced access and limited availability of food grains. In addition to this, some other challenges like Slow delivery of grains, international borders, extreme terrain conditions, natural calamities, harsh weather, and disintegrated storage spaces have to be addressed.

A GUI based executable application was developed, which enables users to select from several working scenarios, and input grain allocations through various government schemes (AAY, PHH, SFY etc). In output, a detailed tagging data is generated giving the amount to be transported through each arc to meet demand at minimal associated cost; Google map plots are generated for quick visualization.

Contents

Nomenclature

Node	Analogous to FPS, Mill, Godown, Warhouse etc.
Arc	Analogous to the route between two locations/nodes
Tagging	This is in the form of a triplet: Node1-Arc-Node2

WFP	--	World Food Programme
FCI	--	Food Corporation of India
TPDS	--	Targeted Public Distribution System
NFSA	--	National Food Security Act
AAY	--	Antyodaya Anna Yojana
PHH	--	Priority Households
SFY	--	State Food Scheme
GUI	--	Graphical User Interface
USN	--	Udham Singh Nagar

List of Figures

Chapter 1 Introduction

United Nations World Food Programme (WFP) is the world's largest humanitarian organization fighting hunger worldwide through delivering food assistance in emergencies and working with communities to improve nutrition and build resilience. WFP in India through its Country Strategic Plan (2019-2023) aims to support India in achieving the targets of Sustainable Development Goals 2 and 17, by enhancing the efficiency, targeting, service delivery and supply chain of government programmes for improving access to food. This project is part of an agreement between WFP, FCI and the Govt. of Uttarakhand to support them to make their supply chain systems more efficient and cost-effective overcoming.

Some of the challenges that the State of Uttarakhand has to address on a regular basis - like earthquakes, fires, etc. are not predictable and frequently result in considerable costs due to emergency solutions that need to be quickly planned and implemented. However, there are also more predictable challenges that tend to happen around the same periods in the year (e.g. seasonal access constraints and monsoons). To reduce cost of transportation and ensure pre-prepositioning of adequate safety stocks in the areas where access is affected by these seasonal and predictable hazards, WFP India shall support the state of Uttarakhand by undertaking a supply chain optimization of the end-to-end TPDS operations including procurement in the state.

The TPDS supply chain of Uttarakhand has never been optimised before now. The taggings in it's network exist in chronological manner i.e. in the order they were added with time and demand, hence, were not optimised. Uttarakhand procures Rice for itself and relies on FCI for supply of wheat. The stages of grain flow -

<u>Rice</u>: Procurement Centre, Mill, Base Godown, Interior Godown, Fair price shops
<u>Wheat:</u> FCI warehouses, Base/ Internal warehouses, Fair price shops

A similar project was executed for the state of Odisha, where a savings of around 30% was observed just by changing allocation (tagging) of different nodes. It was not required to physically relocate any node to yield the optimal solution and stakeholders (farmers and beneficiaries) were not affected.

Chapter 2 Literature Survey

2.1 Network Flow Models

Flow Network is a special case of the more general linear program. It is a very important class since most of the aspects of actual situations can be readily recognized as networks and the representation of the model is much more compact. A directed graph with vertices and edges, denoted by G(N,A), where N is the set of n nodes and A is the set of m directed arcs. Two distinguished nodes, Source (s) and Sink (t) to supply and receive flow. Flows will come out of s and need to obey a few constraints with capacities of arcs in order to reach t. Each arc, (i,j) $\in$ A also has an associated cost per unit flow.

2.2 TPDS Supply Chain Optimization in Odisha

The Odisha Government has made bizarre advancement and enhancements to the Targeted Public Distribution System of the state ever since the adoption of the National Food Security Act' 2013 and the implementation of End to End computerization of the TPDS. As a part of this process, the distribution and procurement systems for TPDS have been digitalised and automated. In an endeavor to make these systems even more transparent, accountable and efficient on the request of the Odisha Government, World Food Programme undertook an assessment of the entire distribution and procurement operations and also the deployed software systems that is the Supply Chain Management System (SCMS) and the Paddy Procurement Automation System (P-PAS), during April, 2017.

2.3 Uttarakhand TPDS

Uttarakhand PDS is taken by the Department of Food, Civil Supplies and Consumer Affairs. Supply to consumers a) Rice & Wheat b) Sugar and Kerosene. Schemes covered a) NFSA (AAY and PHH) b) State Food Scheme (SFY) c) Mid day Meal. FCI godowns in Uttarakhand:174Internal Godowns and 22 Base Godowns, 913 FPSs in the state. Food grains movement in Uttarakhand is mostly via railways and roads, areas with difficult terrain are supplied via roads and extension of helicopter services in areas such as Pitthorgarh. F.E.A.S.T. (Food & Essential Commodities Assurance & Security Target) is a project which evaluates the online allocation quantity of digitized ration card and also maintains the supply chain management of food grains. e-Khareed is an e-governance initiative portal to usher in transparency at all levels in the food grains procurement process. RCMS (Ration Card Management System) is used for digitizing the existing Ration card and allowing users to enter a new ration card and manage the old ration cards.

2.4 COIN-OR

COIN-OR stands for Computational Infrastructure for Operations Research. The goal of this project is to create an open-source community that can help speed up the formulation and deployment of algorithms, models, and cutting-edge computational research in the field of operations research. Research papers and articles in OR journals on mathematical concepts often consist of supplementary numerical outcomes obtained from computational work. The models, data and software implementations which are used for producing the numerical outcomes are, in general, not published. The current researchers in the field of operations research who are willing to reproduce these computational outcomes, draw comparisons and extend the state of the art.

2.5 PuLP

PuLP is an open source software, which is programmed in Python language. It is used to define optimisation problems in the form of mathematical models. PuLP has the option to call any of numerous external Linear Programminig solvers (GLPK, CBC, Gurobi, CPLEX etc) to work out the model and then use various commands and functions of python to manipulate and display the solution. It is very easy to interchange solvers in PuLP and does not require to make many changes in the program, only certain parameters for the LpProblem.solve function need to be changed. PuLP elegantly narrows the gap between a python programmer and an Operations Research practitioner. This helps the OR practitioner to use Python to easily formulate and solve linear programming and integer programming mathematical models and at the same time having access to all the tools present in the python standard library. The python programmer can now quickly incorporate Linear Programming models in complex programs programmed in Python.

2.6 Tkinter

Tkinter bundle is a GUI library in Python. When combined with Tkinter, Python delivers a quick and simple approach to design GUI applications. Powerful object oriented interface is provided to the Tk GUI toolkit by Tkinter. Tkinter is incorporated with Mac OS X, Microsoft Windows and Linux installs of Python. Tkinter is a free open source software, which is released under a Python license

Chapter 3 Project Objectives and Work Plan

3.1 Project Definition and Motivation

Over time, the TPDS of Uttarakhand has observed very few upgrades in terms of efficiency improvement, resulting in a higher cost to be incurred by the government.
This project tries to enhance the efficiency and effectiveness of Govt.'s food safety network for Uttarakhand state TPDS using Computational Infrastructure for Operations Research (COIN-OR). Also, Reduce the cost of transportation and ensure pre-prepositioning of adequate safety stocks in the areas where access is affected by seasonal and predictable hazards.

Adopting a more strategic and long-term approach to plan the allocation of food grains inside the state, supply chain issues can be anticipated, and optimization techniques can be leveraged to evaluate different implementation scenarios that are cost-efficient solutions yet practical and implementable.

To deliver a software tool, which provides features like scenario selection and freedom to the operator to update input and allocation data according to situational need. Ultimately providing the end-user with a yearly supply plan to be followed.

3.2 Objectives of the work

1. Mapping of Supply Chain systems landscape (such as F.E.A.S.T. and e-Khareed) and analysis of the available data and quality to ensure data coverage and quality are appropriate for optimization purposes.
2. Data analysis and mapping to ensure a thorough understanding of the network complexity and key supply chain considerations.
3. Supply chain network mapping, at the end of the phase, submit a visual representation of Uttarakhand’s supply chain network.
4. Design and develop prototype of optimization algorithm to optimize TPDS supply chain operations for all districts of Uttarakhand
5. Creating a user-friendly GUI which on hitting the solve button returns all the taggings in an excel file along with their visualization on a map
6. Finalization of Tool, handover of product and training the concerned officials about the real-time usage of the tool.

3.3 Methodology

The goal of the project is to reduce the total transportation cost and find the resulting node taggings for Uttarakhand. A network flow model is formulated and is modelled using PuLP library in Python. PuLP is an open-source linear programming (LP) package which largely uses Python syntax and comes packaged with many industry-standard solvers. It also integrates nicely with a range of open source and commercial LP solvers. Using PuLP, the minimum cost is obtained along with the taggings.

A user-friendly GUI is created using Tkinter using Python which takes some input values and on hitting the solve button runs the optimization engine in the background and returns the optimal cost, taggings in an excel sheet and visualisation of the obtained taggings on a map. Below is the Gantt chart representing some of the objectives that have been achieved and also the future work plan.

3.4 Gantt Chart

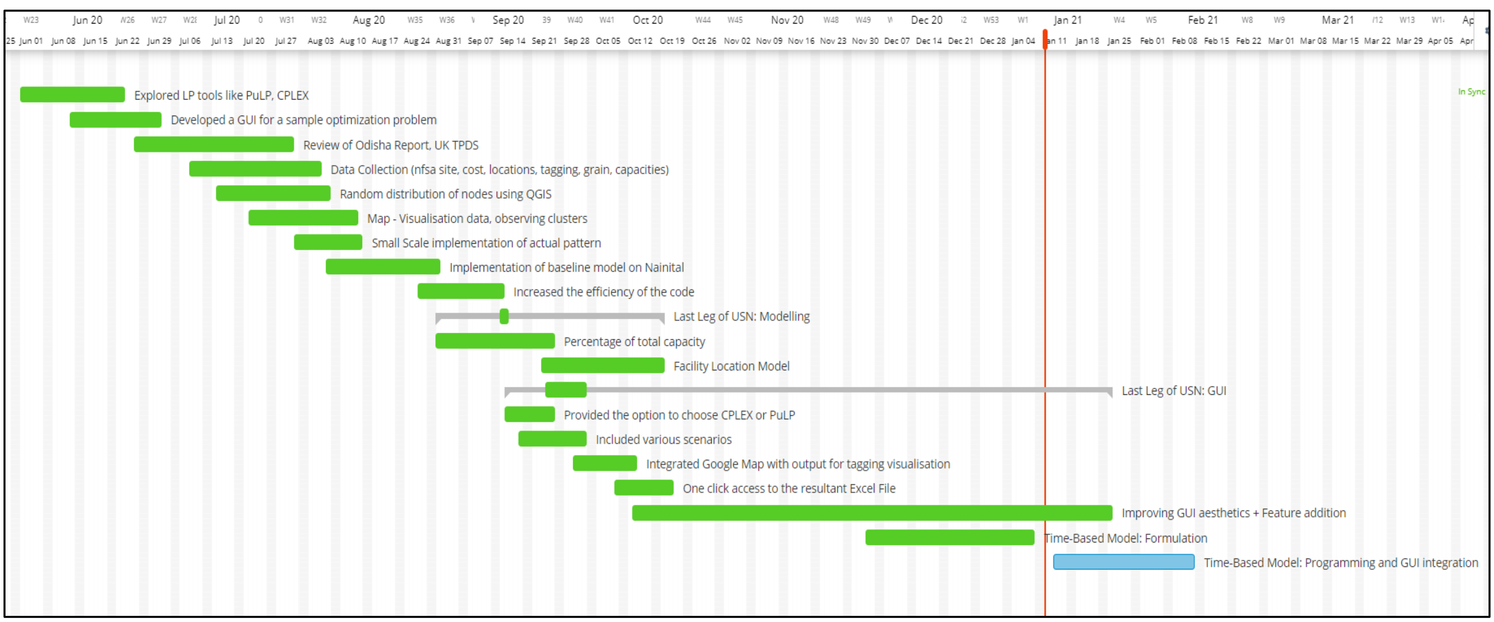


Figure 3.4.1 Gantt Chart

Chapter 4 Work Progress

4.1 Theoretical Model

4.1.1 Baseline Scenario:

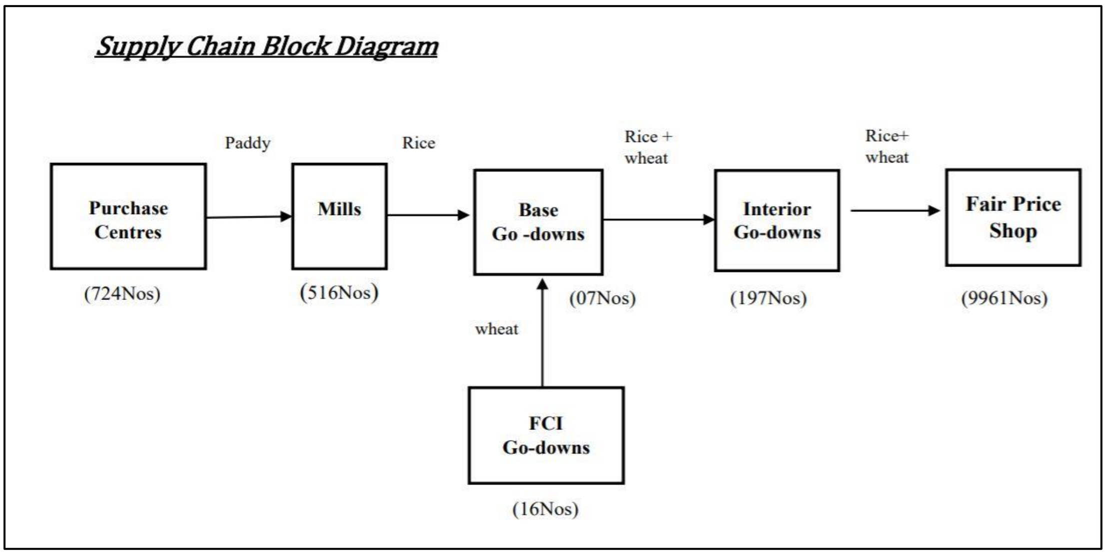


Figure 4.1.1 Network Flow Diagram for Baseline Scenario

C_{ij}^{k} = Cost of transportation from i^{th} node to j^{th} for k^{th} commodity (Rs/Quintal).
X_{ij}^{k} = Volume of food grains transported from i^{th} to j^{th} node for k^{th} commodities (Qtl).

Objective Function:

$$\text{Min } \sum_k \sum_i \sum_j C_{ij}^{k} X_{ij}^{k} \quad \forall (i,j) \in \text{Script\_S}$$

Subject to constraints,

- *Mass – balance (capacity) constraints.*

  $$\sum_j X_{ji}^{k} - \sum_j X_{ij}^{k} \le C_i, \text{ Where } C_i\text{=Capacity of } i^{th} \text{ node}$$

  Also, we have assumed that the grain flow is uni-directional and a node only supplies to its succeeding node. This is part of pre-processing for this problem
  Note: The capacity of Fair price shop is ∞ or large valued number "L".

- *Non -Negatively restriction:*

  $$X_{ij}^{k} \ge 0, C_{ij}^{k} \ge 0$$

- *Demand at FPS:*

$D(f)_j^k$=Demand of j^{th} fair price shop for K^{th} commodity.

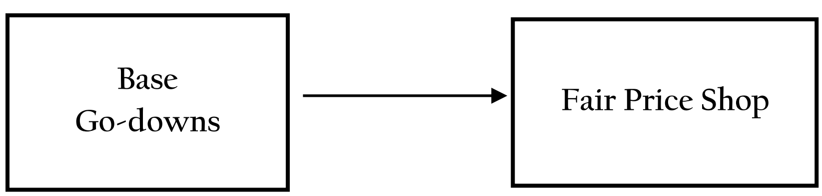
$$\sum_{i=1}^{i=i} X_{ij}^{K} \geq \alpha_j D(f)_j^k$$

i.e. we have to supply at least the amount required at FPS.

4.1.2 Uttarakhand TPDS Problem – USN Last Leg Model:

Let B be the set of base godowns, F_c be the set of FCI godowns each containing 7 warehouses and 409 FPSs respectively. This was gathered from the available data for the district. This is a multi-commodity flow problem having K commodities, where K=1 or 2 for rice and wheat respectively. S is the set of tagging/ arcs between nodes i and j or $(i, j) \in S$. Let C_{ij}^k be the cost of transportation from the i^{th} base godown to j^{th} FPS for the k^{th} commodity.

USN LAST LEG _MODEL



(07 Nos) (409 Nos)

Figure 4.1.2 Network Flow Diagram for Last Leg of USN

$$\min \sum_{k} \sum_{i} \sum_{j} C_{ij}^k x_{ij}^k$$

subject to

$$\sum_{j}^{n} \sum_{k=1}^{2} x_{ij}^k \leq C_i$$

$$\sum_{i}^{m} x_{ij}^k \geq D_j^k$$

$$x_{ij}^k \geq 0$$

where C_i is the capacity of the i^{th} base warehouse, D_j^k is the demand of the j^{th} FPS for the k^{th} commodity

4.1.3 Time Based Model:

Extending our baseline scenario to implement it for the instance of complete year. The result will bear the optimal grain flow amount and tagging details, for the months when the corresponding grain supply is permissible.

Unlike previous scenarios, an inventory needs to be maintained and updated, the behavior of which will depend on the type of harvesting season going on.

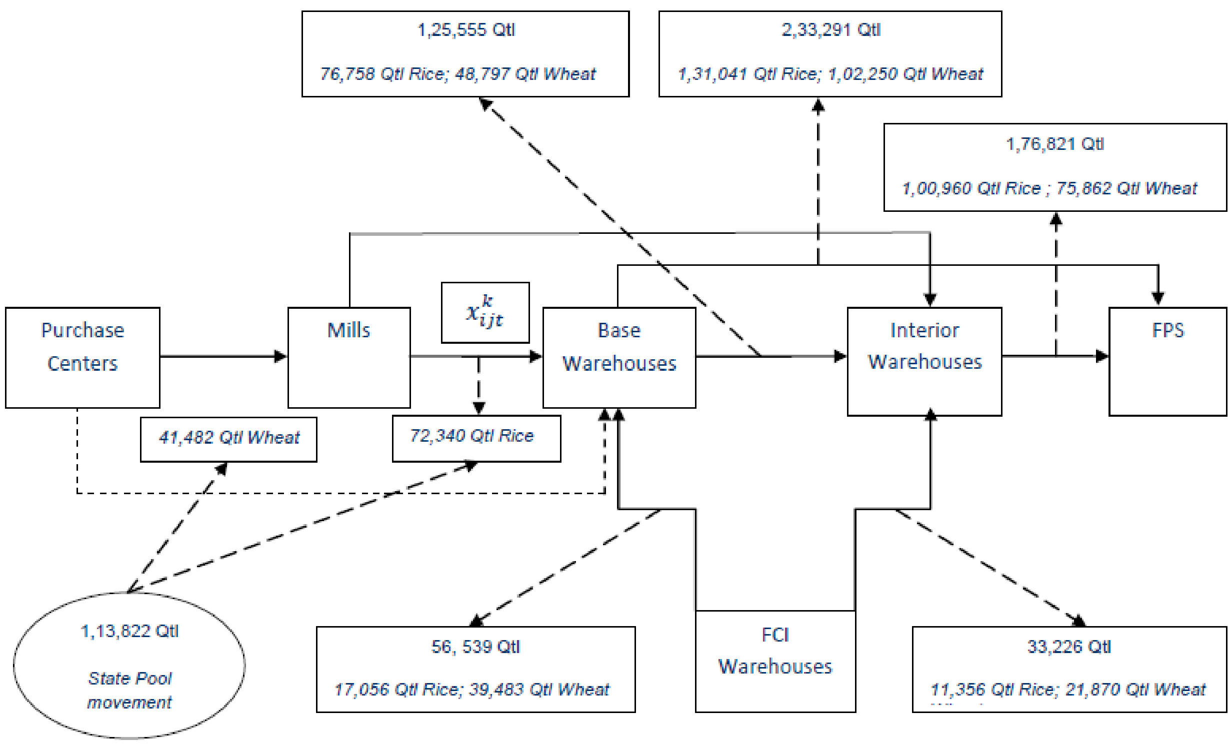


Figure 4.1.3 Network Flow Diagram for Time based model on UK state

Objective Variables:

$F_{it}^{(k)}$ = Supply of grains from ith FCI for kth commodity in tth month (source 2).

k = 1 for rice

k = 2 for wheat

$x_{ijt}^{(k)}$ = Vol. of grain flow (in quintal) from i^{th} to j^{th} node in t^{th} month of k^{th} commodity.

y_{ijt} = Vol. of paddy flow (in quintal) from i^{th} to j^{th} node in t^{th} month, i ∊ PC's & j ∊ Mills.

$c_{ijt}^{(k)}$ = Cartage cost (Rs/quintal) for i^{th} to j^{th} node in t^{th} month of k^{th} commodity.

Objective Function:

$$\text{Min } \sum_i \sum_j \sum_k \sum_t \left(x_{ijt}^{(k)} c_{ijt}^{(k)} \right) + \sum_i \sum_j \sum_t \left(y_{ijt} c_{ijt}^{(1)} \right); \ \forall \text{ (i , j) } \in \text{S (set of arcs) } \forall \text{ t} \in \{1,2,3\text{.....}12\}$$

Applying constraints over constituent category of nodes,

- Purchase/ Procurement Centers:

Assumption – The movement of commodities procured over a month, from i[th] PC to Mills, is made altogether in the next month from the procurement. Any change in the same will be incorporated accordingly

Constraint I. - Updating inventory level of each PC for each month based on inflow, outflow and carry-forward amount

I.A - Updating Paddy inventory stored in i[th] PC, at the end of t[th] month ($IP_{it}^{(1)}$)

$$IP_{i(t+1)}^{(1)} = IP_{it}^{(1)} + \left(P_{i(t+1)}^{(1)} - \sum_{j=1}^{m} y_{ij(t+1)} \right); \text{ i} \in \text{PC's and j} \in \text{Mills; } \forall \text{ t} \in \{0,1,2,3\text{.....}12\}$$

I.B - Updating Paddy inventory stored in i[th] PC, at the end of t[th] month ($IP_{it}^{(2)}$)

$$IP_{i(t+1)}^{(2)} = IP_{it}^{(2)} + \left(P_{i(t+1)}^{(2)} - \sum_{j=1}^{b} x_{ij(t+1)}^{(2)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(2)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(2)} \right);$$

$$\text{i} \in \text{PC's and j} \in \text{BW U IW U FPS; } \forall \text{ t} \in \{0,1,2,3\text{.....}12\}.$$

Constraint II. - Capacity constraints on the storage size of each procurement center (CP_i)

$$IP_{it}^{(1)} + IP_{it}^{(2)} + P_{i(t+1)}^{(1)} + P_{i(t+1)}^{(2)} \leq CP_i \text{ ; i} \in \text{PC's ; } \forall \text{ t} \in \{0,1,2,3\text{.....}12\}$$

Constraint III. - Non-negativity constraint on on-hand inventory of PC at any time "t"

$$IP_{it}^{(1)} \text{ , } IP_{it}^{(2)} \geq 0 \text{ ; } \forall \text{ t} \in \{ 1,2,3\text{.....}12 \} \ \forall \text{ i} \in \text{PC's}$$

Constraint IV. - Incorporating the constraint on procurement due to harvesting season

IV.A – Paddy procurement only occurs during kharif season i.e., the months of October to December

$$IP_{it}^{(1)} \leq 0; \qquad \forall \text{ t} \in \{ 1,2,3,4,5,6,7,8,9 \} \ \forall \text{ i} \in \text{PC's}$$

IV.B – Wheat procurement only occurs during Rabi season i.e., the months of October to December

$$IP_{it}^{(2)} \leq 0; \qquad \forall \text{ t} \in \{ 1,2,3,7,8,9,10,11,12 \} \ \forall \text{ i} \in \text{PC's}$$

- Mills:

Constraint V. – Limitation on the Processing/ Milling capacity of a mill for a given month

$$Z_{it} \leq MC_i \text{ ; } i \in \text{Mills ; } \forall\, t \in \{0,1,2,3\ldots..12\}$$

Z_{it} = Quantity of rice processed by i^{th} Mill by the end of t^{th} month

Constraint VI. – 2 types of grain stored- paddy and processed rice

VI.A - Updating Paddy inventory stored in i^{th} Mill, at the end of t^{th} month ($IM_{it}^{(1)}$).

$$IM_{i(t+1)}^{(1)} = (IM_{it}^{(1)} + \left(\sum_{j=1}^{p} y_{ji(t+1)} - \frac{Z_{i(t+1)}}{CF_i}) \right);$$

$$\forall\ i \in \text{Mills } j \in \text{PC's } \forall t \in \{0,1,2,3\ldots..12\}$$

VI.B - Inventory of processed rice stored in i^{th} Mill, at the end of t^{th} month ($IM_{it}^{(2)}$)

$$IM_{i(t+1)}^{(2)} = (IM_{it}^{(2)} + \left(Z_{i(t+1)} - \sum_{j=1}^{b} x_{ij(t+1)}^{(1)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(1)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(1)} \right);$$

$$\forall\ i \in \text{Mills } j \in \text{PC's BW U IW U FPS } \forall\ t \in \{0,1,2,3\ldots..12\}.$$

Constraint VII. – Limitation on Total storage capacity of Mills

$$(IM_{it}^{(1)} + IM_{it}^{(2)} + \sum_{j=1}^{p} y_{ji(t+1)} \leq (1+\rho_M)CM_i); \forall\ j \in \text{PC's } \forall\ i \in \text{Mills } \forall\ t \in \{0,1,2,3\ldots..11\};$$

CM_i = Storage Capacity (Unprocessed paddy + rice) of i^{th} Mill

Constraint VIII. - Non-negativity constraint on on-hand inventory of Mill at any time "t"

$$IM_{it}^{(1)}, IM_{it}^{(2)} \geq 0; \qquad \forall\ t \in \{1,2,3\ldots..12\}\ \forall\ i \in \text{Mills}$$

Constraint IX. – Limitation on food grain outflow due to Inventory capacity.

$$\sum_{j=1}^{b} x_{ij(t+1)}^{(1)} + \sum_{j=1}^{I} x_{ij(t+1)}^{(1)} + \sum_{j=1}^{f} x_{ij(t+1)}^{(1)} \leq IM_{it}^{(2)} + Z_{i(t+1)} \text{ ;}$$

$$\forall\ i \in \text{Mills } j \in \text{PC's U BW U IW U FPS } \forall\ t \in \{0,1,2,3\ldots..12\}$$

Constraint X. – Ensuring all paddy is processed to rice at end of year

$$\sum_{t=1}^{12} Z_{it} \leq \sum_{j=1}^{p} \sum_{t=1}^{12} CF_i y_{jit} \text{ ;} \qquad i \in \text{Mills, } j \in \text{PC's}$$

Constraint XI. – Constraint on amount of processed rice

$$Z_{i(t+1)} \leq \sum_{j=1}^{p} y_{ji(t+1)} + IM_{it}^{(1)} \text{ ;} \quad \forall\ i \in \text{Mills } j \in \text{PC's } \forall\ t \in \{0,1,2,3\ldots..12\}$$

- <u>Base Warehouse:</u>

Inflow - PC's, Mills, FCI Warehouse Outflow- Internal warehouse, FPS

Constraint XII. – Inventory level of each Base Warehouse at the end of t^{th} month.

XII.A – Inventory of Rice stored in i^{th} Base Warehouse, at the end of t^{th} month

$$IB_{i(t+1)}^{(1)} = IB_{it}^{(1)} + \left(\sum_{j=1}^{m} x_{ji(t+1)}^{(1)} + \sum_{j=1}^{fc} x_{ji(t+1)}^{(1)} + \sum_{j=1}^{b} x_{ji(t+1)}^{(1)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(1)} - \sum_{j=1}^{b} x_{ij(t+1)}^{(1)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(1)} \right);$$

$\forall$ i ϵ BW & j ϵ Mills U FCIW U IW U BW U FPS $\forall$ t ϵ {0,1,2,3…..11}

XII.B – Inventory of Wheat stored in i^{th} Base Warehouse, at the end of t^{th} month

$$IB_{i(t+1)}^{(2)} = IB_{it}^{(2)} + \left(\sum_{j=1}^{p} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{fc} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{b} x_{ji(t+1)}^{(2)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(2)} - \sum_{j=1}^{b} x_{ij(t+1)}^{(2)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(2)} \right);$$

$\forall$ iϵ BW & j ϵ PC's U FCIW U IW U BW U FPS $\forall$ t ϵ {0,1,2,3…..11}

Constraint XIII. - Constraint on overall capacity of node

$$(IB_{it}^{(1)} + IB_{i(t+1)}^{(2)} + \sum_{j=1}^{p} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{m} x_{ji(t+1)}^{(1)} + \sum_{k=1}^{2} \sum_{j=1}^{fc} x_{ji(t+1)}^{(k)} + \sum_{k=1}^{2} \sum_{j=1}^{b} x_{ji(t+1)}^{(k)} \leq (1 + \rho_B)\, CB_i);$$

CB_i = Capacity of i^{th} BW ; $\forall$ i ϵ BW & j ϵ PC's U Mills U FCIW U BW ;

ρ_b = 0.15 (currently) $\forall$t ϵ {0,1,2,3…..11}

Constraint XIV.- No cyclic Transfer i.e., BW to BW transfer not permitted

$$x_{iit}^{(k)} = 0 ; \ \forall \text{ i } \epsilon \text{ BW \& j } \epsilon \text{ BW} ; \forall \text{ k } \epsilon \{ 1, 2 \} ; \forall \text{ t } \epsilon \{ 1,2,3.....12 \}$$

Constraint XV. - Non-negativity constraint on inventory of BW for both commodity at "t"

$$IB_{it}^{(1)}, IB_{it}^{(2)} \geq 0 ; \forall \text{ t} \in \{ 1,2,3.....12 \} \ \forall \text{ i} \in \text{BW}.$$

- Interior/ Internal Warehouse:

Constraint XVI. - Inventory level of each Internal Warehouse at the end of t^{th} month

XVI.A - Inventory of Rice stored in i^{th} IW, at the end of t^{th} month

$$II_{i(t+1)}^{(1)} = II_{it}^{(1)} + \left(\sum_{j=1}^{m} x_{ji(t+1)}^{(1)} + \sum_{j=1}^{b} x_{ji(t+1)}^{(1)} + \sum_{j=1}^{fc} x_{ji(t+1)}^{(1)} + \sum_{j=1}^{I} x_{ji(t+1)}^{(1)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(1)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(1)} \right);$$

$\forall$ i ϵ IW & j ϵ Mills U BW U FCIW U IW U FPS $\forall$ t ϵ {0,1,2,3…..11}

XVI.B - – Inventory of Wheat stored in i^{th} IW, at the end of t^{th} month

$$II_{i(t+1)}^{(2)} = II_{it}^{(2)} + \left(\sum_{j=1}^{p} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{b} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{fc} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{I} x_{ji(t+1)}^{(2)} - \sum_{j=1}^{I} x_{ij(t+1)}^{(2)} - \sum_{j=1}^{f} x_{ij(t+1)}^{(2)} \right);$$

$\forall$ i ϵ IW & j ϵ PC's U BW U FCIW U IW U FPS $\forall$ t ϵ {0,1,2,3…..11}

Constraint XVII. – Constraint on overall capacity of node

$$(II_{it}^{(1)} + II_{it}^{(2)} + \sum_{j=1}^{p} x_{ji(t+1)}^{(2)} + \sum_{j=1}^{m} x_{ji(t+1)}^{(1)} + \sum_{k=1}^{2} \sum_{j=1}^{b} x_{ji(t+1)}^{(k)} + \sum_{k=1}^{2} \sum_{j=1}^{fc} x_{ji(t+1)}^{(k)} + \sum_{k=1}^{2} \sum_{j=1}^{I} x_{ji(t+1)}^{(k)} \leq (1 + \rho_I) CI_i);$$

CI_i = Capacity of i^{th} IW $\forall$ i ϵ IW & j ϵ PC's U Mills U BW U FCIW U IW;

$\forall$t ϵ {0,1,2,3…..11}; ρ_I = 0.15 (currently).

Constraint XVIII. - No cyclic Transfer i.e., IW to IW transfer not permitted

$x_{iit}^{(k)} = 0$; $\forall$ i ϵ IW & j ϵ IW ; $\forall$ k ϵ { 1, 2 } ; $\forall$ t ϵ { 1,2,3…..12 }

Constraint XIX. - Non-negativity constraint on inventory of IW for both commodities

$II_{it}^{(1)}$, $II_{it}^{(2)}$ ≥ 0 ; $\forall$ t ∈ { 1,2,3…..12 } $\forall$ i ∈ IW

- Fair Price Shops:
-

Inflow - BW, IW, Mills and PC's

Constraint XX. – Satisfying demand of j^{th} FPS for t^{th} month. Here we have considered t as an enabling provision, although currently the demand for a given FPS is constant every month.

XX.A - Satisfying demand of Rice

$$(\sum_{i=1}^{m} x_{ijt}^{(1)} + \sum_{i=1}^{b} x_{ijt}^{(1)} + \sum_{i=1}^{I} x_{ijt}^{(1)} \geq D_{jt}^{(1)}) ;$$

$\forall$ i ϵ PC's U Mills U BW U IW & j ϵ FPS; $\forall$ t ∈ { 1,2,3…..12 }.

XX.B - Satisfying demand of Wheat

$$(\sum_{i=1}^{p} x_{ijt}^{(2)} + + \sum_{i=1}^{b} x_{ijt}^{(2)} + \sum_{i=1}^{I} x_{ijt}^{(2)} \geq D_{jt}^{(2)}) ;$$

$\forall$ i ϵ PC's U Mills U BW U IW & j ϵ FPS; $\forall$ t ∈ { 1,2,3…..12 }.

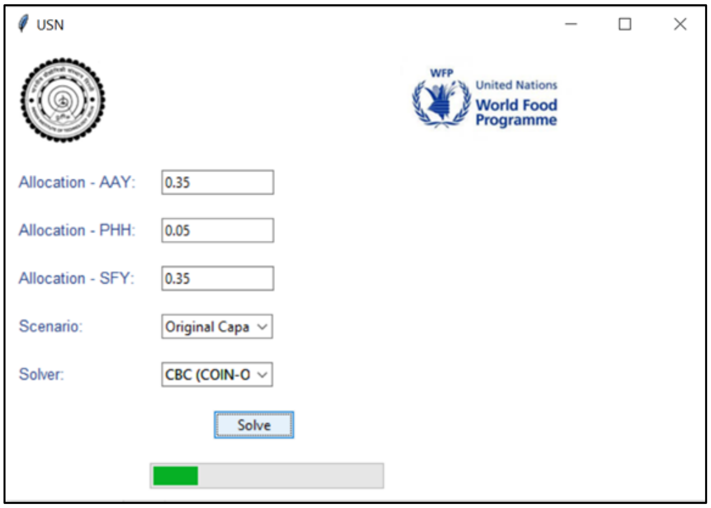
- <u>FCI Warehouse:</u>

Constraint XXI – Constraint on supply amount from FCI godowns

$(\sum_{j=1}^{b} x_{ijt}^{(k)} + \sum_{j=1}^{I} x_{ijt}^{(k)} \leq F_{it}^{(k)})$; $\forall$ k ϵ { 1,2 } ; $\forall$ i ϵ FCIW & j ϵ BW U IW

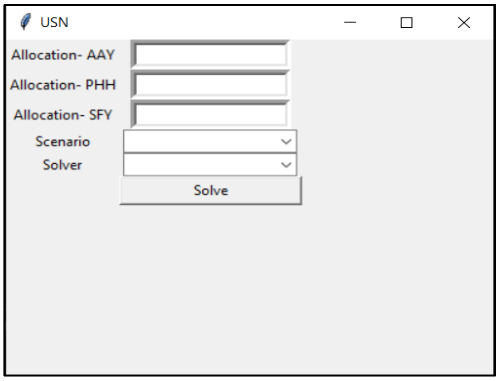
4.2 Software Tool

This is how the main screen looks like on running the executable file. Here we type the values of AAY, PHH, SFY and in the drop down lists we have various options for scenarios and two options for solver (COIN-OR and CPLEX). After entering the values and choosing the desired options, we hit the solve button. The progress bar below indicates the progress of the optimization engine running in the background



On hitting the solve button, we get the minimum cost and two output files which can be accessed directly by clicking the respective buttons. On clicking the map button, a visualization of the obtained taggings can be seen on a map. Further, on clicking the excel button, we obtain the node to node tagging data corresponding to the optimal cost in an excel file

4.2.1 GUI Modifications over Time



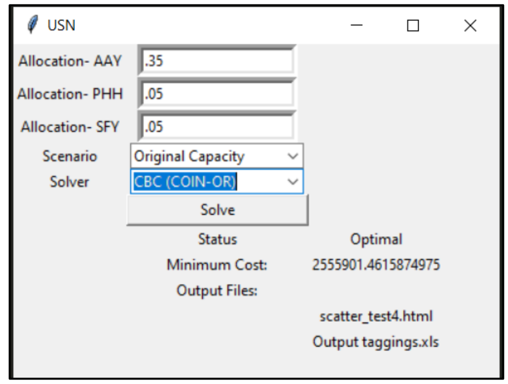


Figure 4.2.1.1 Initial GUI

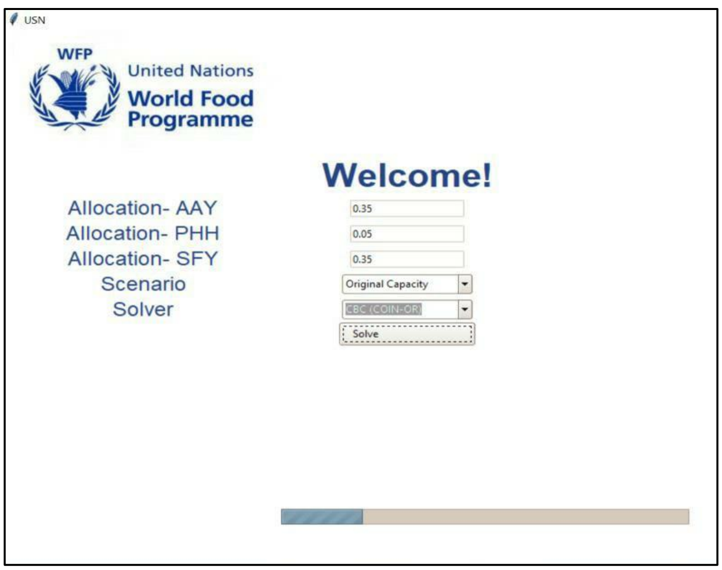


Figure 4.2.1.2 GUI Design at intermediate iteration

We analysed many Windows based GUIs and decided to work on the same lines, thus designing an interface with which the user is more likely to be familiar.

In the final iteration of the software tool, all the labels and buttons and the format of the tool has been made similar to standard windows GUI. We also made sure that all the widgets in the tool are aligned properly with adequate space. In this version, it is also possible to recompute the Minimum Cost without shutting down our tool by simply changing the allocation values and then hitting on the solve button again. Other features that we have added include displaying important info when the cursor hovers over any button and the option to copy the location of the output files by right clicking on the buttons.

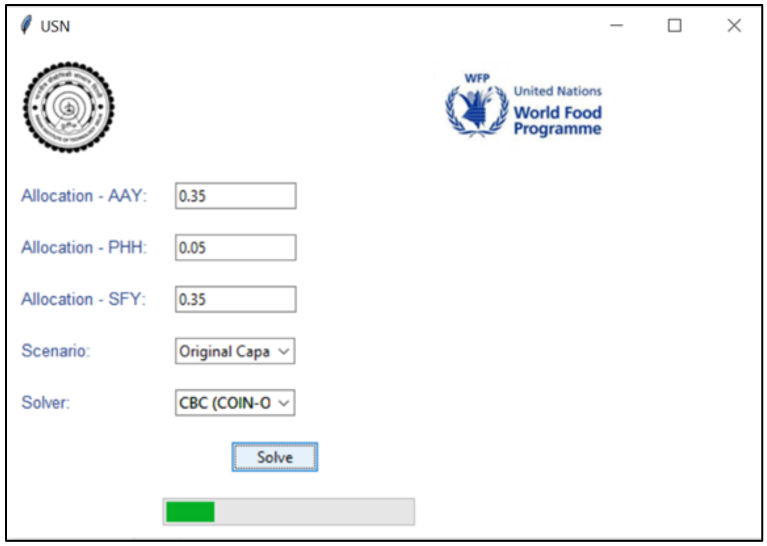
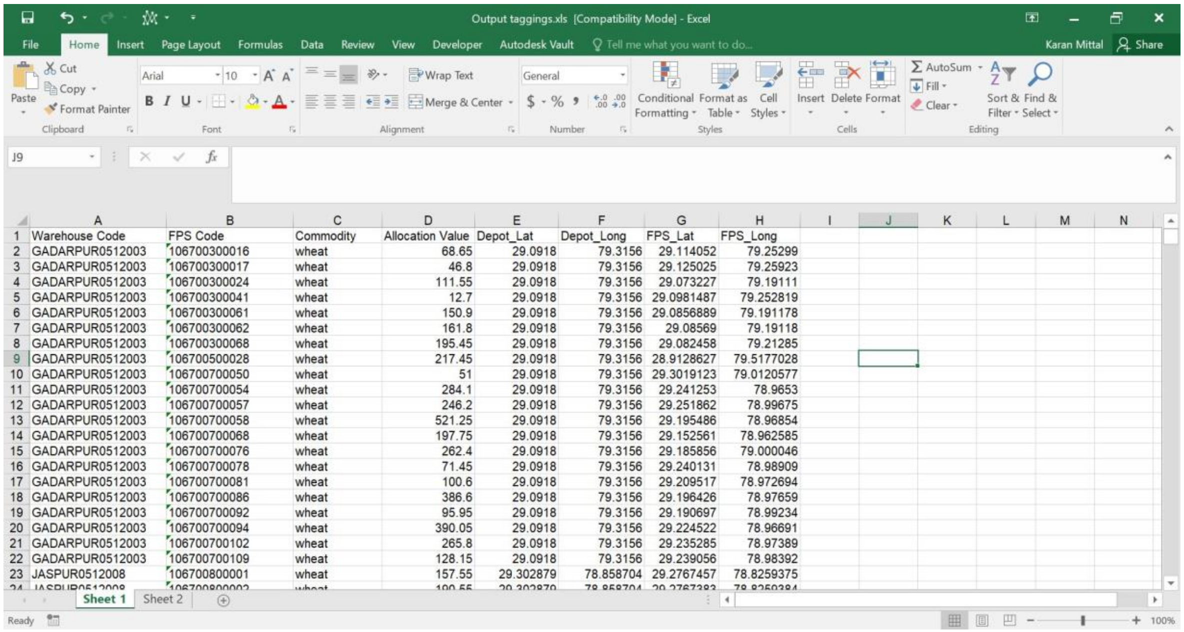


Figure 4.2.1.3 GUI Design at Present

4.2.2 Tagging

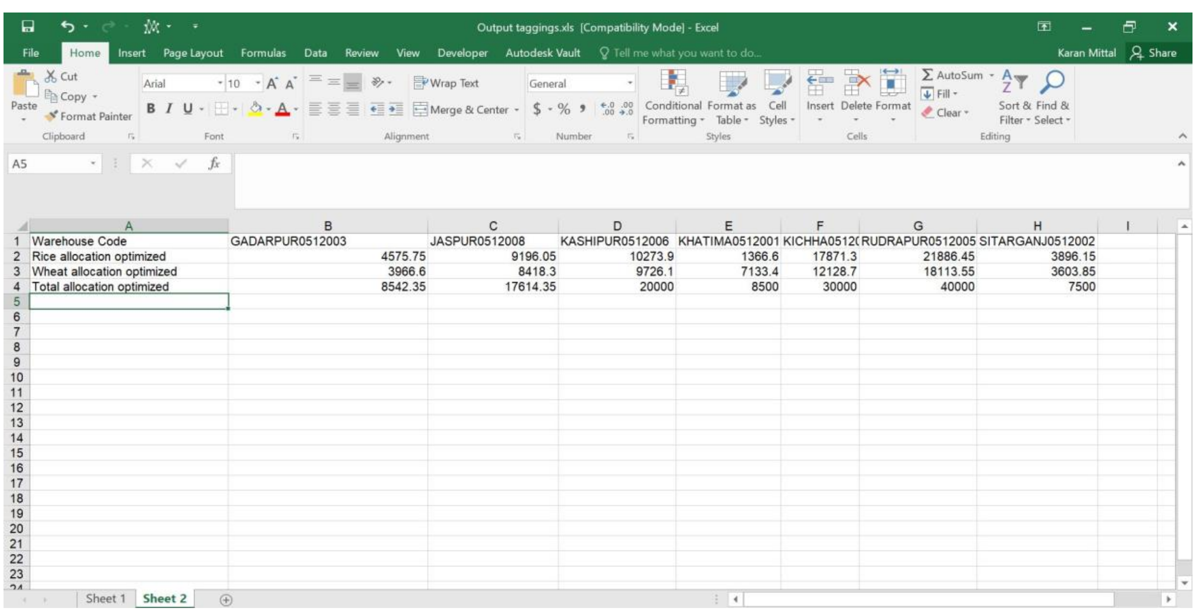
Excel file containing all the obtained taggings along with their respective coordinates and allocation value



	Warehouse Code	FPS Code	Commodity	Allocation Value	Depot_Lat	Depot_Long	FPS_Lat	FPS_Long
2	GADARPUR0512003	106700300016	wheat	68.65	29.0918	79.3156	29.114052	79.25299
3	GADARPUR0512003	106700300017	wheat	46.8	29.0918	79.3156	29.125025	79.25923
4	GADARPUR0512003	106700300024	wheat	111.55	29.0918	79.3156	29.073227	79.19111
5	GADARPUR0512003	106700300041	wheat	12.7	29.0918	79.3156	29.0981487	79.252819
6	GADARPUR0512003	106700300061	wheat	150.9	29.0918	79.3156	29.0856889	79.191178
7	GADARPUR0512003	106700300062	wheat	161.8	29.0918	79.3156	29.08569	79.19118
8	GADARPUR0512003	106700300068	wheat	195.45	29.0918	79.3156	29.082458	79.21285
9	GADARPUR0512003	106700500028	wheat	217.45	29.0918	79.3156	28.9128627	79.5177028
10	GADARPUR0512003	106700700050	wheat	51	29.0918	79.3156	29.3019123	79.0120577
11	GADARPUR0512003	106700700054	wheat	284.1	29.0918	79.3156	29.241253	78.9653
12	GADARPUR0512003	106700700057	wheat	246.2	29.0918	79.3156	29.251862	78.99675
13	GADARPUR0512003	106700700058	wheat	521.25	29.0918	79.3156	29.195486	78.96854
14	GADARPUR0512003	106700700068	wheat	197.75	29.0918	79.3156	29.152561	78.962585
15	GADARPUR0512003	106700700076	wheat	262.4	29.0918	79.3156	29.185856	79.000046
16	GADARPUR0512003	106700700078	wheat	71.45	29.0918	79.3156	29.240131	78.98909
17	GADARPUR0512003	106700700081	wheat	100.6	29.0918	79.3156	29.209517	78.972694
18	GADARPUR0512003	106700700086	wheat	386.6	29.0918	79.3156	29.196426	78.97659
19	GADARPUR0512003	106700700092	wheat	95.95	29.0918	79.3156	29.190697	78.99234
20	GADARPUR0512003	106700700094	wheat	390.05	29.0918	79.3156	29.224522	78.96691
21	GADARPUR0512003	106700700102	wheat	265.8	29.0918	79.3156	29.235285	78.97389
22	GADARPUR0512003	106700700109	wheat	128.15	29.0918	79.3156	29.239056	78.98392
23	JASPUR0512008	106700800001	wheat	157.55	29.302879	78.858704	29.2767457	78.8259375

Figure 4.2.2.1 Optimal Tagging suggestions produced by GUI

Sheet 2 contains the rice, wheat and total allocation corresponding to each warehouse in Udham Singh Nagar



	Warehouse Code	GADARPUR0512003	JASPUR0512008	KASHIPUR0512006	KHATIMA0512001	KICHHA0512(	RUDRAPUR0512005	SITARGANJ0512002
2	Rice allocation optimized	4575.75	9196.05	10273.9	1366.6	17871.3	21886.45	3896.15
3	Wheat allocation optimized	3966.6	8418.3	9726.1	7133.4	12128.7	18113.55	3603.85
4	Total allocation optimized	8542.35	17614.35	20000	8500	30000	40000	7500

Figure 4.2.2.2 Overall grain flow through each warehouse

4.2.3 Map

Used gmplot library to plot the obtained taggings on the map. Below is the plot for taggings of the last leg of Udham Singh Nagar.

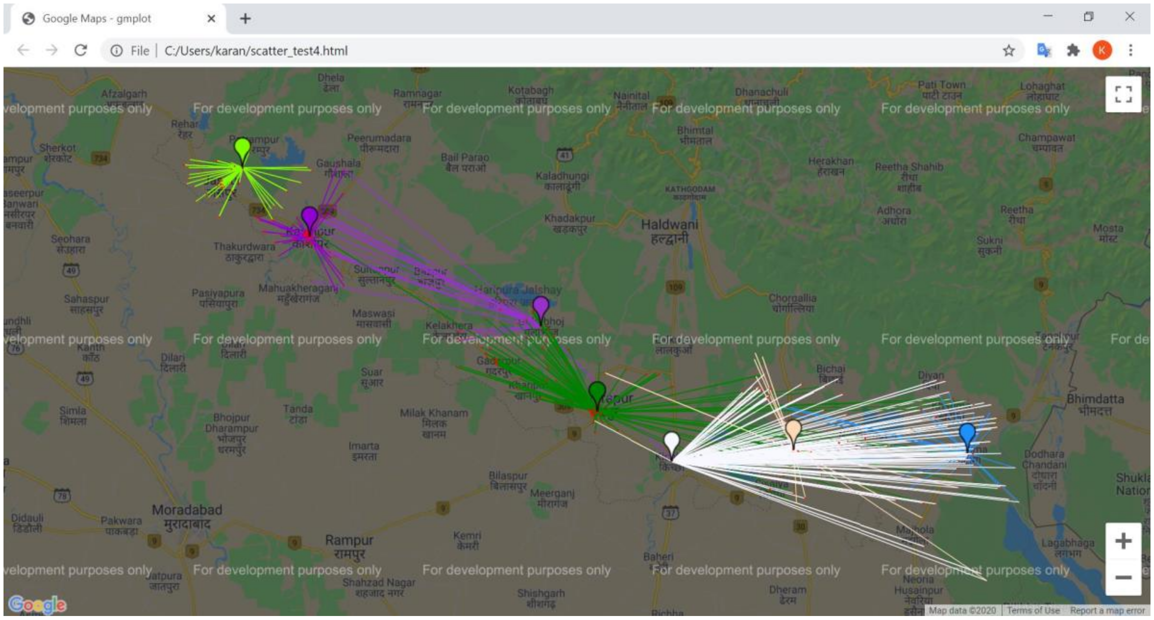


Figure 4.2.3.1 Tagging's plotted using gmplot Library

Similar Plot obtained using Basemap library of Matplotlib

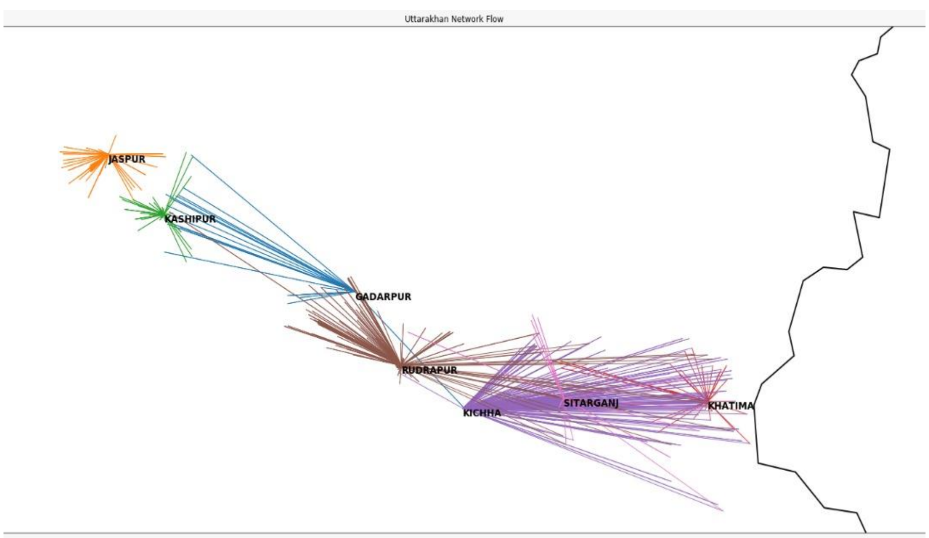


Figure 4.2.3.2 Tagging's plotted using basemap Library

Using these libraries for visualization purpose can substitute paid software like Tableau and ArcGIS, serving the same purpose

Conclusions and Further Work

Conclusions

Using Linear Programming techniques, minimum transportation cost for the last leg of Udham Singh Nagar was obtained. Optimal routes from Warehouses to Fair Price Shops were also obtained indicating drastic reduction in the total transportation cost compared to the original taggings. The obtained taggings were visualised on a map for a better understanding.

A user-friendly GUI was created that can be used by any government official to get the optimal taggings. The official simply needs to input the grain allocation values through various government schemes (AAY, PHH, SFY etc) and select the scenario and solver from the dropdown list. Now the official needs to click on the solve button and in a few seconds, he/she will get the optimal taggings. Further we were able to formulate the Time-based model after several iterations of refining.

Further Work

The future work will aim at extending the model to all legs of the supply chain and further extend it to all the districts of Uttarakhand. In order to accomplish this, following are the future goals:

- Programming the time – based model and designing a GUI for the same
- Implementing the time - based model on all districts of Uttarakhand
- Further improvement of the GUI by making it more user-friendly and easier to use for a government official

References

1. S.Mitchell, ‘An Introduction to pulp for Python Programmers’,The Python Papers Monograph, Vol. 1 (2009)

2. D G Mogale, Mukesh Kumar, et al., ’Grain silo location-allocation problem with dwell time for optimization of food grain supply chain network’, Transportation Research Part E: Lo- gistics and Transportation Review, Volume 111, March 2018, Pages 40-69

3. D G Mogale, Sri Krishna Kumar, Manoj Kumar Tiwari, ’An MINLP model to support the movement and storage decisions of the Indian food grain supply chain’, Control Engineering Practice, Volume 70, January 2018, Pages 98-113

4. Alexandre Dolgui, D.G.Mogale, Rishabh Kandhway, Sri Krishna Kumar, Manoj Kumar Tiwari, ’A multi-period inventory transportation model for tactical planning of food grain supply chain’, Computers and Industrial Engineering, Volume110, August2017, Pages 379-304
5. Nasrin Asgari, Reza Zanjirani Farahani, Hannaneh Rashidi-Bajgan, Mohsen S Sajadieh, ’Developing model-based software to optimise wheat storage and transportation:A real-world application’, Applied Soft Computing Journal, 13 (2) (2013), pp. 1074-1084

6. L.M. Maiyar, J.J. Thakkar, ’A combined tactical and operational deterministic food grain transportation model: Particle swarm based optimization approach’, Computers & Industrial Engineering, 110 (2017), pp. 30-42