Norwegian School of Economics

Bergen, Autumn, 2020

Optimization and Cost Allocation in Collaborative Transportation

Potential Savings and Decreased CO^2 Emissions from Optimized Collaboration in Fuel Distribution in Norway

Morten Lid

Supervisor: Roman Kozlov

Master Thesis, Master of Science in Economics and Business Administration, Business Analytics (BAN)

NORWEGIAN SCHOOL OF ECONOMICS

Preface

This thesis was written as a part of the Master of Science in Economics and Business Administration, Business Analytics, at Norwegian School of Economics (NHH), autumn 2020.

I took many very interesting courses at NHH and one of my absolutely favorites was Professor Mario Guajardo's course Decision Modelling in Business (BAN402) at autumn 2019. Here we learned the data tool A Mathematical Programming Language (AMPL) and mathematical methods for cost allocation, which are the two main tools I uses in this paper.

The work in this thesis has been very interesting and also challenging particularly regarding data collection.

I would like to thank supervisor Professor Roman Kozlov for enthusiastic and fruitful discussions, and for adapted accessibility in the challenging coronavirus situation.

I would like to thank Eirik Aronsen for highly needed information regarding fuel depots in the analyses.

Bergen, December 20, 2020

Morten Lid

Abstract

The purpose of this study is to find potential savings and potential reduction in CO^2 emission from an optimized collaboration in distributing fuel in Norway, from fuel depots to gas stations, compared to a non-collaborative fuel distribution. In addition to this the purpose is to make an optimization program in A Mathematical Programming Language (AMPL) for collaborating in distributing fuel in Norway, which can be easily adjusted and expanded regarding input data.

The answers to these questions I aim to find in this thesis are in my opinion very interesting these days of increased focus on effectiveness and the problems regarding global warming from CO^2 emission.

The possible collaborations in this thesis includes the 4 fuel companies Circle K, Shell (ST1), Esso and YX/Uno-X, with their supply points from in total 20 fuel depots to their demand points at gas stations at all cities and places in Norway which is inhabited by 10000 citizens or more.

To get answers to these questions I will do several analyses and calculations. I will process data, create and run several analysis programs in AMPL. From the cost results of full collaboration (grand coalition) I will allocate costs to each company in the mathematic cost allocation method Shapley Values. Then I will from methods in cooperative game theory and results from Shapley Values and other results in AMPL find out if the grand coalition is the most beneficial coalition for each company, or if there are smaller coalitions which can be more beneficial. I will then also find the potential reduction in CO^2 emission from a optimized collaboration.

My findings from the analyses and calculations, given the input data, show that a optimized collaboration in the grand coalition is most beneficial for all the 4 companies and that there is a lot to save for each company from such full collaborative fuel distribution in Norway, compared to non-collaborative distribution. Further my findings, given the input data, show that the potential total decreased CO^2 emission from the trucks from full collaboration is considerable.

Contents

1. Introduction

I wanted to analyze and optimize a real-world problem in the data tool A Mathematical Programming Language (AMPL) where the results touch a topic that matters at a considerable level. After considering different topics I decided to analyze and optimize a collaborative distribution of fuel from depots to gas stations in Norway, to find potential savings and decreased CO^2 emissions compared to a non-collaborative distribution. This includes to find out if all the gas companies have highest benefits from joining full collaboration, where all four companies collaborate together in one coalition (grand coalition), or if there are smaller coalitions which can be more beneficial.

By creating a optimizing program in the data tool AMPL and using the cost allocation method Shapley Values including condition tests I can find potential savings in total and for each gas company, and find the potential amount of decreased CO^2 emission from the trucks from a optimized collaborative fuel distribution, compared to a non-collaborative distribution. I think this is an important subject to look into, these days with increased demands on efficiency and the problems regarding global warming due to CO^2 emission.

The thesis with all its analyses and calculations are limited to involve the four main gas companies in Norway; Circle K, Shell (ST1), Esso and YX/UNO-X. Further it is limited to include the gas stations cities and places in Norway which is inhabited by 10000 citizens or more, which in total are 59 cities/places. The thesis will include all the 21 fuel depots in Norway. The analyses is based on distribution of fuel with conventional tank trucks with diesel engines. The possibility of using electric tank trucks is not considered here, as I believe that this, if happens, at least will be several years from now, due to the need of driving long distances with extremely heavy loads, which electric vehicles is not able to do today. This is also what others, which has look into it, believes (Spilde & Skotland, 2016). I will therefore analyze the situation like it is today and in my believe is likely to be in a long period of time.

The return routes of the trucks are not included in the analyses as it is not clear if the trucks always are returned to the supply point in every cases or if they can be hired and therefore not returned to the supply points. The results from the analyses is therefore values for one way transportation, which I see as the core operations in distributing fuel. From this one can understand that the savings and decreased CO^2 emissions is likely to be even larger than the results show. However, the percentage savings and percentage reduced CO^2 emissions will probably be not much affected from this.

According to Norwegian Competition Authority the four fuel companies Circle K, Shell, Esso and YX can pic up fuel from each other's depots through agreements (Johansen, 2010). This means that these fuel companies can share all depots with each other if they want, without breaking the law. However, the degree of collaboration is not clear, due to difficulties finding clear information on this topic. However, I see it as there is a high probability that distribution of fuel in Norway today is not optimized toward 100% optimization. From this we can not find the potential savings and potential decreased CO^2 emissions from a optimized collaboration in fuel distribution, compared to the collaboration as it is today. However, we can try estimate the potential savings and potential decreased CO^2 emission from a optimized collaboration in fuel distribution, compared to a non-collaborative fuel distribution in Norway to get pointers of the amounts of savings and reduced CO^2 emissions optimization can give, and try find out if a coalition which consists all the four companies are most beneficial or if smaller coalitions can be more beneficial. Maybe finding these results also can shred light on potentials and further encourage to optimization. A optimization program in AMPL could may also be a help in further optimization in fuel distribution in Norway.

The distances between depot and gas stations are possible to find, using updated data tools, I will here use Google Maps for this (Google, 2005).

When it comes to demand from gas stations cities/places we can estimate these data. The same goes for depot capacities, but the latter are more uncertain. I will therefore run three series of analyses in AMPL, three series of corresponding cost allocations and three series of corresponding condition tests. That is, three different approaches regarding depot capacities to get more reliable conclusions from the results.

The distribution of fuel, which includes both gas and diesel, is in this study merged to one, as to analyze these two fuels separately would only give minimal changes to the results. As mentioned already depot capacities are uncertain, then it would be meaningless to analyze gas and diesel separately as the results already will probably have way more inaccuracy from uncertain depot capacities compared to the inaccuracy from analysing gas and diesel as one.

The cost for transporting fuel is also possible to estimate, I have calculated this from own knowledge and statements from workers in the industry.

I will in this thesis aim to find reliable results and pointers for what potential savings and decreased CO^2 emissions are possible from a optimized collaboration in fuel distribution in Norway, which also includes to find out if a optimized collaboration involves all the 4 main gas companies or if smaller coalitions are more beneficial.

I will in this thesis also aim to make a AMPL program which is suitable and easy to use regarding adjustments, changed and added data for use in collaborative fuel distribution in Norway.

Below the introduction chapter there will be a theoretical chapter, where I introduce the building blocks needed for the analyses and calculations.

In the methods chapter I will explain how I collected, calculated and used the data, explain how I created and run the AMPL program and show how I calculated and allocated the costs in Shapley Values. Further I will show how I checked these allocated costs for several conditions. This chapter is followed by a second methods chapter where I will show how I created new modified files in AMPL to run several different analyses. There will also come explanations regarding further changes in the AMPL files and calculations in the results chapter, to make the thesis easier to follow for the reader.

I will then in the results chapter run all the analyses and do all the corresponding calculations, including cost allocations and corresponding condition tests and show results at each section. As mentioned I will in this chapter explain some of the changes I will be doing along the different analyses and calculations to make the thesis easier to follow for the reader.

Then I will discuss the findings in the discussion chapter.

I will finally write conclusions in the conclusion chapter.

2. Theory

In this thesis I want to find potential savings and decreased CO^2 emissions from a optimized collaborative distribution of fuel from depots to gas stations in Norway, compared to a non-collaborative fuel distribution, which includes to find out if all the gas companies have most benefits from joining a full collaboration (grand coalition).

To accomplish the analyses and calculations for finding answers to these questions I needed information, analysis tools and methods. I will here give a introduction of these building blocks which I needed in my analyses and calculations, as a understanding of what they are. This will involve information blocks and concepts, analysis program and calculation methods. However, the technical parts of how I collected, processes and uses these blocks, concepts and tools will come in the methodical chapters.

2.1 Gas stations

2.1.1 Locations

The four companies Circle K, Shell (ST1), Esso and YX/UNO-X has gas stations which are located in very many cities and places in Norway. Most cities and larger places have all four companies' gas stations represented. However, the locations of the gas stations is mostly located at cities and places which has citizens of a certain size, there are not so many gas stations in the middle of nowhere. This thesis is limited to the cities/places in Norway which have citizens of 10000 or more, which are 59 cities/places (Thorsnæs, 2019).

All the 59 cities/places are listed below:

Oslo

Bergen

Stavanger

Trondheim

Fredrikstad

Drammen

Porsgrunn

Kristiansand

Ålesund

Moss

Haugesund

Tønsberg

Sandefjord

Arendal

Bodø

Tromsø

Hamar

Halden

Larvik

Kongsberg

Askøy

Molde

Harstad

Gjøvik

Lillehammer

Horten

Jessheim

Ski

Mo I Rana

Kristiansund

Korsvik

Tromsdale

Hønefoss

Alta

Elverum

Stjørdalshalsen

Askim

Narvik

Leirvik

Osøyro

Råholt

Drøbak

Grimstad

Vennesla

Nesoddtangen

Steinkjer

Bryne

Kongsvinger

Kopervik

Knarrvika

Egersund

Ålgård

Lommedalen

Mandal

Ås

Brummunddal

Førde

Levanger

Konnerud

2.1.2 Demand

Every gas station's city/place has a annual- and statistical demand of fuel. These is mostly corresponding to the size of the population in the city/place.

2.2 Depots

2.2.1 Locations

The gas stations are located throughout Norway at smaller and larger places and cities.

The 4 companies Circle K, Shell (ST1), Esso and YX/UNO-X owns in total 20 fuel depots in Norway and drives fuel from depots to their gas stations in Norway. In addition Equinor has a depot at Mongstad, but this company do not own or operate any gas stations, which the other four companies does (E. Aronsen, personal communication, 2020). Fuel depots are often located near the sea, but not all.

In the following I have noted Shell (ST1) as Shell and YX/Uno-X as YX, due to that I think most people are most familiar to just Shell and just YX. I think the paper now will be easier to follow for the reader.

The 21 depot's locations in Norway are listed below (E. Aronsen, personal communication, 2020):

Cicle K

Cities/places for Circle K's depots in Norway:

Alta

Harstad

Trondheim

Førde

Kristiansand

Oslo

Shell

Cities/places for Shell's depots in Norway:

Tananger

Vestervika

Skjelnan

Kirkenes

Balsfjord

Larsgården

Lillesund

Sjursøya

Esso

Cities/places for Esso's depots in Norway:

Trondheim

Bergen

Slagen

Fredrikstad

YX

Cities/places for YX's depots in Norway:

Mo I Rana

Stavanger

In addition Equinor has a depot at Mongstad at their oil refinery, however Equinor do not own or operate any gas stations, so this depot is only included in three extra analyses in the cases where all four companies collaborate. This is because all the other 45 analyses have to be done without Equinor, thus all analyses for the calculation in Shapley Value formula has to be done without Equinor's depot, to get a correct calculation.

2.2.2 Capacities

The depots have different capacities in the tank at their depots, every depot is not equal regarding volume in the depot's tank. However, the depots get refilled, so in practice the volume of the tank at each depot is not the true capacity.

2.3 Distances

Distances is the distances in km the fuel trucks which distributes the fuel from the depots to the gas stations has to drive. These are one distance per delivering operation, which in these analyses is 1180 distances(routes). The return routes are not included in the analyses.

2.4 Coalitions

The companies can choose to stand alone or to collaborate in smaller or larger coalitions. The larger the coalition the more the companies can expect to save, if all the companies do contribute to the collaboration. A company that do not contribute to the collaboration and therefore should not be in the collaborative coalition is e.g. a company that operates only at regions where none of the other companies operates, hence there can not be created savings from this company.

In game theory, a coalition where all the players (companies) are represented is called the grand coalition (Coalition Theory Network, 2020).

2.5 AMPL

A Mathematical Programming Language (AMPL) is a computational mathematical programming tool. AMPL is a tool which is typically used for optimization problems in for example transportation, shipping, oil refinery or electricity markets, where one wants to optimize the routes in transporting and shipping, optimize the flows in a oil refinery or optimize the power management in electricity markets.

A major advantage of the program is that it can handle a high amount of input data (Robert Fourer, 2003). This comes to use in tasks were one have large scale problems (Havås, Alfred, Jim, & Mirjam, 2013), like for example very many routes to chose from or very many constrains to fulfill.

For each analysis one want to run in AMPL, the program need four files, which are the model file, the data file, the run file and the text file.

The model file is usually created without numbers, but contains the minimizing or maximization function and conditions, where these are linked to the numbers in the data file. Plus different coding for the program.

The data file is created with sets and parameters, with often very many parameters and large tables of numbers.

The run file is coded for deciding which model file one wants to use in the analysis, which data file and what(results) to display in which text file.

2.6 Game theory

2.6.1 Shapley Values

Shapley Values is a well-known mathematical method for allocating cost or payoff from a coalition to the participants (players), were all these players collaborating. Shapley Values has a formula for calculating the players allocated cost or payoff. One can either put all possible coalition costs (including stand-alone costs) into the formula and get the result for the current player’s allocated cost. Or one can put coalition payoffs (including stand-alone costs) into the formula and get the result for the current player’s allocated payoff. The formula and mechanism is the same in the two cases.

The purpose in cost allocation is to allocate the coalition’s (grand coalition) cost to each player in a fair way, so each player gets their cost reduced (from collaborating) according to how much they contribute to the collaboration. The sum of the allocated costs is equal to the coalition (grand coalition) cost.

The purpose in payoff allocation is to give each player payoff according to how much they contribute to the collaboration.

Technically the cost in cost allocation is allocated to a player according to the average marginal cost the coalition gets at the moment this player enters the coalition, were the coalition is formed one by one player (Centon, 2019). To find the average marginal cost the grand coalition gets of a specific player one must calculate the marginal cost the coalition gets the moment this player enter the coalition, and calculate this in all possible order(sequences) the grand coalition can be formed. Then one need to sum all these costs and divide the sum by the number of cases. The marginal cost of this player in each of these cases is the cost of the coalition as it is when this player has entered the coalition minus the cost of the coalition right before this player entered the coalition. There can also be none or one player in the "coalition" at the moment right before this player enter this "coalition".

For payoff allocation it will be the same mechanism, were Shapley Values finds the average increased savings (payoff) the coalition gets of this player.

To get a better understanding of Shapley Values calculation I will show the mechanism in a simple example of three players. Here I will also prove that Shapley Values can be used both for payoff allocation and for cost allocation.

Imagine we have the three players A, B and C, collaborating in a coalition, were all three players collaborating, and that they have reduced the total cost due to the collaborating.

I will show the calculations for player A:

We have in this example the optimized costs and optimized payoff for all possible coalitions these three players can form, inclusive stand-alone costs:

Coalition	Optimized cost	Payoff
A:	11	0
B:	12	0
C:	13	0
AB:	18	5
AC:	10	14
BC:	22	3
ABC:	19	17

All possible sequences (order) the coalition (grand coalition) can be formed and calculation of the marginal cost of player A:

Order		Marginal costs
ABC	(A – 0)	11 – 0 = 11
ACB	(A – 0)	11 – 0 = 11
BAC	(AB – B)	18 – 12 = 6
BCA	(ABC – BC)	19 – 22 = -3
CAB	(AC – C)	10 – 13= -3
CBA	(ABC – BC)	19 – 22 = -3
		= 19

19/6 = 3.167

This means that the allocated cost from Shapley Values to player A is 3.167, which is the average marginal cost of player A.

All possible sequences (order) the coalition (grand coalition) can be formed and calculation of the increased savings (payoff) of player A:

Order		Increased payoff
ABC	(0 – A)	0 – 0 = 0
ACB	(0 – A)	0 – 0 = 0
BAC	(AB – B)	5 – 0 = 5
BCA	(ABC – BC)	17 – 3 = 14
CAB	(AC – C)	14 – 0 = 14
CBA	(ABC – BC)	17 – 3 = 14
		= 47

47/6 = 7.833

This means that the allocated payoff from Shapley Values to player A is 7.833

Payoff for player A should also be the same as; stand alone cost for A minus player A's allocated cost from Shapley Values. Lets check:

11 – 3.167 = 7.833 (yes)

This proves that the Shapley Value can be used for both cost allocation and for payoff allocation.

Shapley Values also have a formula which can be used to calculate Shapley Values (allocated cost or payoff). However, there exist different versions, which of course gives the same results. The most common version seems to be the following version (Cotra, 2019):

$$\varphi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(n - |S| - 1)!}{n!} (v(S \cup \{i\}) - v(S))$$

Where n is the total number of players and S is the subsets of N not containing player i.

Shapley Value (allocated cost) for player A:

((2/6) x (A)) + ((1/6) x (AB-B+AC-C)) + ((2/6) x (ABC-BC))

= ((2/6) x (11)) + ((1/6) x 18-12+10-13)) + ((2/6) x (19-22))

= 3.667 + 0.5 + (-1)

= 3.167 (yes)

2.6.2 Conditions

According to cooperative game theory a coalition’s cost allocation belongs to the core if the cost allocation is based on a coalition where there are no other smaller coalitions that can be more beneficial in the game. Also according to game theory a cost allocation is stable if none of the players are more beneficial to break out to form smaller coalitions or stand alone. A cost allocation can belong to the core and still not be stable, this is usually happening when the cost allocation is not fair regarding to how much saving each player gets compared to the company’s contribution to the collaboration.

To find out if a cost allocation is stable and belongs to the core the cost allocation will first be checked regarding the two conditions, the individual rational- and the coalition rational condition. Finally the summed allocated costs will be checked regarding the efficiency condition.

The individual rational condition is to check whether each player’s allocated cost is lower than the corresponding stand-alone cost. If this is the case for all the players the individual rational condition is fulfilled.

The coalition rational condition is to check if each possible coalition’s (not stand alone and grand coalition) summed allocated costs is lower than that coalition’s optimized cost. If this is the case for each possible coalition, the coalition rational condition is fulfilled.

The efficiency condition is to check if the summed allocated costs is equal to the grand coalition cost. This is simply to check if all costs are allocated and to check that no further costs is added.

3. Creating and run the AMPL program

3.1.1 Coalitions

The companies can distribute fuel alone as stand alone companies or they can form coalitions of two-, three- or four players(companies) to collaborate in distributing fuel. To do all the analyses I need to do in this thesis to find the answers I seek to find I need to investigate all the coalitions the 4 companies can form, plus the 4 players coalition including Equinor’s depot. To allocate the costs from the 4 players coalition (grand coalition) I need the optimized cost results from AMPL for each possible coalition the 4 companies Circle K, Shell, Esso and YX can form. All these analyses will be done in three series, due to three different approaches regarding depot capacities.

All possible coalitions the companies can form

I have listed all the possible coalitions the 4 companies can form below, these are both stand alone and collaborative oriented. For the last and fully collaborated(grand coalition) case I have run 2 different analyses, where the last will include Equinors depot at Mongstad. But as mentioned the latter is not included in the cost allocation formula I uses. However I have used the collaboration situation when Equinor is included when I calculates the second analysis for potential savings and what amount of decreased CO^2 emission is possible, from a perfect collaboration compared to the cases when the companies stand alone and do not collaborate at all.

Circle K

Shell

Esso

YX

Circle K / Shell

Circle K / Esso

Circle K / YX

Shell / Esso

Shell / YX

Esso / YX

Circle K / Shell / Esso

Circle K / Shell / YX

Circle K / Esso / YX

Shell / Esso / YX

Circle K / Shell / Esso / YX

We can see that this is in total 15 different coalitions, including stand alone configurations for the 4 companies. As I wish to find and analyze what collaborations are interesting from the companies point of view regarding cost and to find possible savings which also leads to finding potential emission reductions I have done 45 different analysis in AMPL, plus the 3 analyses which includes Equinors depot at Mongstad at the full collaborative cases, plus 5 analyses for CO^2 emission. However the 45 analyses are 15 x 3 analyses where the only differences in the 3 series is the depot capacities, due to uncertain numbers. By analyszing using these 3 different approaches I will shred more light on the final results and get more reliable conclusions.

3.2 AMPL program

To make the AMPL program I first made one model file that is used unchanged for all the 53 analyses, and for each of the 53 analyses I made one unique data file, one unique run file and one unique text file. So in total 160 files in AMPL. Since I made the model file with no numbers, it allows me to not change anything in the model file when I want to change any numbers for the data. This makes it a lot easier to adjust the program regarding changed data for doing more analysis later on.

Note that all the green text in the program is not parts of the coding, only explanations for the coding.

An overview of the naming of all the 160 AMPL files is given in the appendix.

3.3 Model file

Here I explain and show the model file I have created, which when made with no numbers can be used unchanged through all the 53 different analyses. As you will see I have used no numbers in the model file, because all numbers which may change to different analyses, I want to keep in the data files. The program will then be user friendly and easier to change for further analyses.

3.3.1 Mathematical formulation

Mathematical formulation of objective function and constrains:

$$min\ z = \sum_{i \in I} \sum_{j \in J} k_{ij}\, c_{ij} f_{ij}$$

s.t (1)

$$\sum_{j \in J} f_{ij} \leq s_i\,, \qquad i \in I$$

(2)

$$\sum_{i \in I} f_{ij} = d_j\,, \qquad j \in J$$

$$f_{ij} \geq 0\,, \qquad i \in I, j \in J$$

3.3.2 Model file

The model file have several components, which will be connected to the values and names in the data file trough the run file.

“set I” to include all the names for all “I”, which is the depots.

“set J” to include all the names for all “J”, which is the cities/places.

“param k” to tell the program that “k” is the table of all values for all “I” to all “J”, which is all the 1180 distances from depots to cities/places.

“param s” to tell the program that ”s” is all values for all “I”, which will be the corresponding restrictions for max supply for each depot.

“param d” to tell the program that ”d” is all values for all “J”, which will be the corresponding restrictions for required demand for each city/place.

“param c” to tell the program that ”c” is a constant, which will be the cost constant used in the formula.

“var f” tells the program that the quantity (litres of fuel in 1000 litre) can not be negative.

“minimize z” tells the program to make a minimized solution for “z” from the formula below it, given restrictions.

“Sum” followed by the formula below “minimize z” tells the program what to minimize, which is to multiply the values for the routes “I” to “J” it chooses to use with the cost constant “c” multiplied with the corresponding quantity it chooses to use for the corresponding route “I” to “J”. It will tell the program that “z” is the sum of all this series of sums it uses.

“subject to” tells the program that what is coded below “subject to” is restrictions which the solution has to fulfill when the program minimizes “z”.

The first restriction tells the program that each “I” can be equal to or less than the corresponding “s”, which is the corresponding max supply for each depot.

The second restriction tells the program that each “J” must be equal to the corresponding “d”, which is the corresponding required demand for each city/place.

Figure 3.3.2.1. Model file

Model.mod

```
set I; # set of supply points (depots places)
set J; # set of demand points (gas station places)

param k{I,J}; # distances in km between depots and gas stations
param s{I}; # supply quantity in 1000 litres fuel
param d{J}; # demand quantity in 1000 litres fuel
param c; # cost per 1000 litre km

var f{I,J} >=0; # 1000 litres fuel transported from I to J

minimize z:

    sum{i in I, j in J} (k[i,j]*c*f[i,j]); /* minimizes the total cost, given the restrictions below */

subject to
maxsupply {i in I}:
    sum{j in J} f[i,j] <= s[i]; # max supply of fuel in 1000 litre

requireddemand {j in J}:
    sum{i in I} f[i,j] = d[j]; /* litres of fuel in 1000 litre required from
    gas station places */
```

3.4 Data file

3.4.1 Gas stations locations

I have in the analyses included places/cities for gas stations which holds 10000 citizens or more only, specifically 59 places/cities. All these places/cities I found at Store norske leksikon (Thorsnæs, 2019). When considering this I assume all the 4 companies have gas stations at all these places/cities. This assumption is based on own observations and research, where I found that this assumption seems to hold. If I had included considerable smaller places in Norway this would not be the case, as a small population of course will not give a large enough market for several gas stations, there will some places in Norway only be three, two, one or no gas stations.

The reason why I set the limit at 10000 is that I think this will give strong indications of the information I seek to find. There are values of another parameter in the model that are estimated, since the true values are missing due to difficulties to procure them. Regarding this parameter I thinking of the missing of true capacities for the depots. This parameter will probably create much higher levels of inaccuracy than the missing of smaller places in the model, especially places in the south part of Norway, due to larger and closer populations. Therefore I think it would be meaningless to create the model more accurate regarding the population limit mentioned above at this level of investigation. Another aspect of the depot capacities is that the gas companies probably can easily adjust these capacities to some extent, because they constantly is filled and are therefore probably in practice nearly impossible to give accurate values in the data files.

I will in this study and all its analyses therefore try to find patterns and results which is only accurate to a certain level. Then it would be meaningless to include all places with all gas stations in Norway in these analyses. I still believe to find patterns and results which will give answers at an reasonable level of realistic accuracy.

However, if the program in the future should be changed with more accurate input data, as accurate capasities of the depots and accurate demands for each place/city of each company, it would give more meaning to include more places with a smaller number of citizens.

3.4.2 Depot locations

The locations(cities/places) for the 4 companies 20 depots in Norway depot I have collected from communication with a person (E. Aronsen, personal communication, 2020). This communication was done after difficulty to find updated and reliable information regarding depot locations from research, as I wanted to have reliable depot locations as they are today, to make analyses of good quality.

3.4.3 Distances

All the distances from depots to gas stations cities/places I collected using the tool Google Maps (Google, 2005). This is 1180 distances which I plotted directly into the AMPL program (data files).

From my experience and perception from people in the society Google Maps is a well-known and trusted tool to find the best route to drive when one planning to drive from one destination to a another given destination. I have used this tool myself a lot in my leisure time for years and I trust it a lot. My experience is also that this tool is updated very rapidly to new roads and changes in the roads in Norway. I also find this tool very user friendly and effective to use. These distances I believe is very accurate to reality, as they are based on roads for cars and are based on the routes from one location to another location which are best suitable regarding time used for the ride, which often are the absolutely shortest route or at least one of the shortest routes depending on the quality of the road.

Based on this evaluation I therefore chose Google Maps to find all the 1180 distances I used in the AMPL program.

3.4.4 Depot capacities

I tried to find information on depot capacities, but this seemed difficult and maybe impossible to find information from research. The information regarding depot capacities may also be confidential (E. Aronsen, personal communication, 2020).

Therefore the depot capacities at each depot in the analyses is not accurate to reality, only estimated. I have therefore run three series analyses with three different approaches regarding depot capacities, as I do not have access to accurate data.

However, it would probably be difficult to determine an actual capacity due to the fact that depots are being refilled. The depots are being refilled either from the company that picked up fuel from the depot or from the company that owns the depot (E. Aronsen, personal communication, 2020). As I see it the actual capacity depends mostly on the capacity of the delivery system to the depots. Because of uncertain depot capacities the results in this analyses are not absolutely correct, but I still believe the analysis results will be not far from the truth and give pointers which probably are correct to reality.

Here in the first serie of analyses I have given all depots the equal capacity which is the total annual consumption of fuel in Norway divided by the total number of depots multiplied with approximately 1.5 to give overcapacities at a middle extent.

Total annual fuel consumption in Norway in 1000 litre: 4874893

Total number of fuel depots in Norway including Mongstad: 21

4874893 / 21 x 1.5 = 348206.6

Depot annual capacities in 1000 litre for each depot I have then rounded to: 350000

3.4.5 Gas station demand

The required demands of fuel for each city/place I have set up I expect is a good estimate as I have calculated these data from collecting data for the amount of citizens at each city/place from the updated Store Norske Leksikon (Thorsnæs, 2019) and statistic total annual fuel consumption in Norway from the well-known Statistisk Sentralbyrå (Statistisk Sentralbyrå, 2020). Calculating each cities/places demand includes a lot of numbers and calculations as the numbers change from what coalition I going to analyze. All this calculations I have done in the excel file Thesis Data, and further copied and pasted several different number series into the data files of the corresponding analyses in AMPL.

Figure 3.4.5.1. Excel file

	A	B	C	D	E	F	G	H
1		Annual fuel consumption in 1000 litre for Norv	Depots company/places	Depots supply divic	Depot Companic	Depots max supply eq	Depots max supply limitl	Depots max supply share adjus
2	Gas for vehicles	1028358	CIALT Circle K/Alta	232137.8	Circle K	350000.0	5000000.0	316868.0
3	Diesel for vehicle	2928062	CIHAR Circle K/Harstad	232137.8	Shell	350000.0	5000000.0	237651.0
4	Diesel	918473	CITRH Circle K/Trondheir	232137.8	Esso	350000.0	5000000.0	329055.3
5	Sum	4874893	CIFOR Circle K/Førde	232137.8	YX	350000.0	5000000.0	365617.0
6			CIKRI Circle K/Kristiansar	232137.8	Equinor	350000.0	5000000.0	312297.8
7			CIOSL Circle K/Oslo	232137.8				
8			SHTAN Shell/Tananger	232137.8				
9			SHVES Shell/Vesterviks	232137.8				
10			SHSKJ Shell/Skjelnan	232137.8				
11			SHKIR Shell/Kirkenes	232137.8				
12			SHBAL Shell/Balsfjord	232137.8				
13			SHLAR Shell/Larsgården	232137.8				
14			SHLIL Shell/Lillesund	232137.8				
15			SHSJU Shell/Sjursøya	232137.8				
16			ESTRH Esso/Trondheim	232137.8				
17			ESBER Esso/Bergen	232137.8				
18			ESSLA Esso/Slagen	232137.8				
19			ESFRE Esso/Fredrikstad	232137.8				
20			YXMOI YX/Mo i Rana	232137.8				
21			YXSTA YX/Stavanger	232137.8				
22			EQMON Equinor/Mongst	232137.8				
23			Sum	4874893.0				

Ark1

Figure 3.4.5.2. Excel file

I	J	K	L	M	N	O
Cities/places for gas stati	Citizens for cities/plac	Annual fuel consumption share for cities/places		Annual fuel consumption in 1000 litre for cities/places		Required annual demand in 1000 litre for Circl
Oslo	1019513	32.716 %	OSL	1594850	OSL	518326
Bergen	257082	8.250 %	BER	402160	BER	130702
Stavanger	225020	7.221 %	STA	352004	STA	114401
Trondheim	186364	5.980 %	TRH	291534	TRH	94748
Fredrikstad	113622	3.646 %	FRE	177742	FRE	57766
Drammen	107930	3.463 %	DRA	168838	DRA	54872
Porsgrunn	93255	2.992 %	POR	145881	POR	47411
Kristiansand	64057	2.056 %	KRI	100206	KRI	32567
Ålesund	53254	1.709 %	ALE	83307	ALE	27075
Tønsberg	52419	1.682 %	TON	82000	TON	26650
Moss	47135	1.513 %	MOS	73734	MOS	23964
Haugesund	45040	1.445 %	HAU	70457	HAU	22899
Sandefjord	44368	1.424 %	SAN	69406	SAN	22557
Arendal	43515	1.396 %	ARE	68072	ARE	22123
Bodø	41720	1.339 %	BOD	65264	BOD	21211
Tromsø	40471	1.299 %	TRO	63310	TRO	20576
Hamar	27947	0.897 %	HAM	43718	HAM	14208
Halden	25708	0.825 %	HAL	40216	HAL	13070
Larvik	24647	0.791 %	LAR	38556	LAR	12531
Kongsberg	22219	0.713 %	KON	34758	KON	11296
Askøy	22088	0.709 %	ASO	34553	ASO	11230
Molde	21103	0.677 %	MOL	33012	MOL	10729
Harstad	21070	0.676 %	HAR	32960	HAR	10712
Gjøvik	20589	0.661 %	GJO	32208	GJO	10468
Lillehammer	20580	0.660 %	LIL	32194	LIL	10463
Horten	20504	0.658 %	HOR	32075	HOR	10424
Jessheim	20016	0.642 %	JES	31312	JES	10176
Ski	19546	0.627 %	SKI	30576	SKI	9937
Mo i Rana	18899	0.606 %	MOI	29564	MOI	9608
Kristiansund	18273	0.586 %	KRU	28585	KRU	9290
Korsvik	17981	0.577 %	KOR	28128	KOR	9142
Tromsdalen	17377	0.558 %	TRD	27183	TRD	8835
Hønefoss	17055	0.547 %	HON	26680	HON	8671
Alta	15342	0.492 %	ALT	24000	ALT	7800
Elverum	15117	0.485 %	ELV	23648	ELV	7686
Stjørdalshalsen	14723	0.472 %	STJ	23032	STJ	7485
Askim	14488	0.465 %	ASK	22664	ASK	7366
Narvik	14148	0.454 %	NAR	22132	NAR	7193
Leirvik	14126	0.453 %	LEI	22098	LEI	7182
Osøyro	13911	0.446 %	OSO	21761	OSO	7072
Råholt	13504	0.433 %	RAH	21125	RAH	6866
Drøbak	13393	0.430 %	DRO	20951	DRO	6809
Grimstad	13304	0.427 %	GRI	20812	GRI	6764
Vennesla	13118	0.421 %	VEN	20521	VEN	6669
Nesoddtangen	13076	0.420 %	NES	20455	NES	6648
Steinkjer	12985	0.417 %	STE	20313	STE	6602
Bryne	12202	0.392 %	BRY	19088	BRY	6204
Kongsvinger	12034	0.386 %	KOV	18825	KOV	6118
Kopervik	11561	0.371 %	KOP	18085	KOP	5878
Knarrvika	11502	0.369 %	KNA	17993	KNA	5848
Egersund	11433	0.367 %	EGE	17885	EGE	5813
Ålgård	11335	0.364 %	ALG	17732	ALG	5763
Lommedalen	11200	0.359 %	LOM	17520	LOM	5694
Mandal	10909	0.350 %	MAN	17065	MAN	5546
Ås	10868	0.349 %	AAS	17001	AAS	5525
Brumunddal	10660	0.342 %	BRU	16676	BRU	5420
Førde	10339	0.332 %	FOR	16174	FOR	5256
Levanger	10333	0.332 %	LEV	16164	LEV	5253
Konnerud	10314	0.331 %	KOU	16134	KOU	5244
Sum	3116292	100.000 %		4874893		1584340

Figure 3.4.5.3. Excel file

P	Q	R	S	T	U	V
le K	Required annual demand in 1000 litre for Shell		Required annual demand in 1000 litre for Esso		Required annual demand in 1000 litre for	Sum Required annual demand in 1000 litre for Circle K/Shell/Ess
OSL	518326	OSL	358841	OSL	199356	
BER	130702	BER	90486	BER	50270	
STA	114401	STA	79201	STA	44001	
TRH	94748	TRH	65595	TRH	36442	
FRE	57766	FRE	39992	FRE	22218	
DRA	54872	DRA	37988	DRA	21105	
POR	47411	POR	32823	POR	18235	
KRI	32567	KRI	22546	KRI	12526	
ALE	27075	ALE	18744	ALE	10413	
TON	26650	TON	18450	TON	10250	
MOS	23964	MOS	16590	MOS	9217	
HAU	22899	HAU	15853	HAU	8807	
SAN	22557	SAN	15616	SAN	8676	
ARE	22123	ARE	15316	ARE	8509	
BOD	21211	BOD	14684	BOD	8158	
TRO	20576	TRO	14245	TRO	7914	
HAM	14208	HAM	9837	HAM	5465	
HAL	13070	HAL	9049	HAL	5027	
LAR	12531	LAR	8675	LAR	4819	
KON	11296	KON	7820	KON	4345	
ASO	11230	ASO	7774	ASO	4319	
MOL	10729	MOL	7428	MOL	4126	
HAR	10712	HAR	7416	HAR	4120	
GJO	10468	GJO	7247	GJO	4026	
LIL	10463	LIL	7244	LIL	4024	
HOR	10424	HOR	7217	HOR	4009	
JES	10176	JES	7045	JES	3914	
SKI	9937	SKI	6880	SKI	3822	
MOI	9608	MOI	6652	MOI	3696	
KRU	9290	KRU	6432	KRU	3573	
KOR	9142	KOR	6329	KOR	3516	
TRD	8835	TRD	6116	TRD	3398	
HON	8671	HON	6003	HON	3335	
ALT	7800	ALT	5400	ALT	3000	
ELV	7686	ELV	5321	ELV	2956	
STJ	7485	STJ	5182	STJ	2879	
ASK	7366	ASK	5099	ASK	2833	
NAR	7193	NAR	4980	NAR	2767	
LEI	7182	LEI	4972	LEI	2762	
OSO	7072	OSO	4896	OSO	2720	
RAH	6866	RAH	4753	RAH	2641	
DRO	6809	DRO	4714	DRO	2619	
GRI	6764	GRI	4683	GRI	2601	
VEN	6669	VEN	4617	VEN	2565	
NES	6648	NES	4602	NES	2557	
STE	6602	STE	4570	STE	2539	
BRY	6204	BRY	4295	BRY	2386	
KOV	6118	KOV	4236	KOV	2353	
KOP	5878	KOP	4069	KOP	2261	
KNA	5848	KNA	4048	KNA	2249	
EGE	5813	EGE	4024	EGE	2236	
ALG	5763	ALG	3990	ALG	2216	
LOM	5694	LOM	3942	LOM	2190	
MAN	5546	MAN	3840	MAN	2133	
AAS	5525	AAS	3825	AAS	2125	
BRU	5420	BRU	3752	BRU	2084	
FOR	5256	FOR	3639	FOR	2022	
LEV	5253	LEV	3637	LEV	2021	
KOU	5244	KOU	3630	KOU	2017	
	1584340		1096851		609362	4874893.0

Figure 3.4.5.4. Excel file

W	X	Y	Z	AA	AB	AC	AD
sso/YX	Required annual demand in 1000 litre for Circle K/Shell		Required annual demand in 1000 litre for Circle K/Esso		Required annual demand in 1000 litre for Circle K/YX		Required annual demand in 1000 litre for Shell/E
OSL	1036652	OSL	877167	OSL	717682	OSL	877167
BER	261404	BER	221188	BER	180972	BER	221188
STA	228803	STA	193602	STA	158402	STA	193602
TRH	189497	TRH	160344	TRH	131190	TRH	160344
FRE	115532	FRE	97758	FRE	79984	FRE	97758
DRA	109744	DRA	92861	DRA	75977	DRA	92861
POR	94823	POR	80235	POR	65647	POR	80235
KRI	65134	KRI	55113	KRI	45093	KRI	55113
ALE	54149	ALE	45819	ALE	37488	ALE	45819
TON	53300	TON	45100	TON	36900	TON	45100
MOS	47927	MOS	40554	MOS	33181	MOS	40554
HAU	45797	HAU	38751	HAU	31706	HAU	38751
SAN	45114	SAN	38173	SAN	31233	SAN	38173
ARE	44247	ARE	37439	ARE	30632	ARE	37439
BOD	42421	BOD	35895	BOD	29369	BOD	35895
TRO	41151	TRO	34820	TRO	28489	TRO	34820
HAM	28417	HAM	24045	HAM	19673	HAM	24045
HAL	26140	HAL	22119	HAL	18097	HAL	22119
LAR	25061	LAR	21206	LAR	17350	LAR	21206
KON	22593	KON	19117	KON	15641	KON	19117
ASO	22459	ASO	19004	ASO	15549	ASO	19004
MOL	21458	MOL	18157	MOL	14855	MOL	18157
HAR	21424	HAR	18128	HAR	14832	HAR	18128
GJO	20935	GJO	17714	GJO	14494	GJO	17714
LIL	20926	LIL	17707	LIL	14487	LIL	17707
HOR	20849	HOR	17641	HOR	14434	HOR	17641
JES	20352	JES	17221	JES	14090	JES	17221
SKI	19875	SKI	16817	SKI	13759	SKI	16817
MOI	19217	MOI	16260	MOI	13304	MOI	16260
KRU	18580	KRU	15722	KRU	12863	KRU	15722
KOR	18283	KOR	15470	KOR	12658	KOR	15470
TRD	17669	TRD	14951	TRD	12232	TRD	14951
HON	17342	HON	14674	HON	12006	HON	14674
ALT	15600	ALT	13200	ALT	10800	ALT	13200
ELV	15371	ELV	13006	ELV	10642	ELV	13006
STJ	14971	STJ	12667	STJ	10364	STJ	12667
ASK	14732	ASK	12465	ASK	10199	ASK	12465
NAR	14386	NAR	12173	NAR	9959	NAR	12173
LEI	14363	LEI	12154	LEI	9944	LEI	12154
OSO	14145	OSO	11969	OSO	9793	OSO	11969
RAH	13731	RAH	11619	RAH	9506	RAH	11619
DRO	13618	DRO	11523	DRO	9428	DRO	11523
GRI	13528	GRI	11446	GRI	9365	GRI	11446
VEN	13339	VEN	11286	VEN	9234	VEN	11286
NES	13296	NES	11250	NES	9205	NES	11250
STE	13203	STE	11172	STE	9141	STE	11172
BRY	12407	BRY	10498	BRY	8590	BRY	10498
KOV	12236	KOV	10354	KOV	8471	KOV	10354
KOP	11755	KOP	9947	KOP	8138	KOP	9947
KNA	11695	KNA	9896	KNA	8097	KNA	9896
EGE	11625	EGE	9837	EGE	8048	EGE	9837
ALG	11526	ALG	9752	ALG	7979	ALG	9752
LOM	11388	LOM	9636	LOM	7884	LOM	9636
MAN	11092	MAN	9386	MAN	7679	MAN	9386
AAS	11051	AAS	9351	AAS	7650	AAS	9351
BRU	10839	BRU	9172	BRU	7504	BRU	9172
FOR	10513	FOR	8895	FOR	7278	FOR	8895
LEV	10507	LEV	8890	LEV	7274	LEV	8890
KOU	10487	KOU	8874	KOU	7261	KOU	8874
	3168680		2681191		2193702		2681191

Figure 3.4.5.5. Excel file

AE	AF	AG	AH	AI	AJ	AK	AL
Esso	Required annual demand in 1000 litre for Shell	/YX	Required annual demand in 1000 litre for Esso	/YX	Required annual demand in 1000 litre for Circle K/Shell/	Esso	Required annual demand in 1000 litre for Circle K/Shel
OSL	717682	OSL	558197	OSL	1335493	OSL	1236008
BER	180972	BER	140756	BER	351890	BER	311674
STA	158402	STA	123202	STA	308004	STA	272803
TRH	131190	TRH	102037	TRH	255092	TRH	225939
FRE	79984	FRE	62210	FRE	155524	FRE	137750
DRA	75977	DRA	59093	DRA	147733	DRA	130849
POR	65647	POR	51058	POR	127646	POR	113058
KRI	45093	KRI	35072	KRI	87680	KRI	77660
ALE	37486	ALE	29157	ALE	72893	ALE	64563
TON	36900	TON	28700	TON	71750	TON	63550
MOS	33181	MOS	25807	MOS	64518	MOS	57144
HAU	31706	HAU	24660	HAU	61650	HAU	54604
SAN	31233	SAN	24292	SAN	60730	SAN	53790
ARE	30632	ARE	23825	ARE	59563	ARE	52755
BOD	29369	BOD	22842	BOD	57106	BOD	50579
TRO	28489	TRO	22158	TRO	55396	TRO	49065
HAM	19673	HAM	15301	HAM	38253	HAM	33882
HAL	18097	HAL	14075	HAL	35189	HAL	31167
LAR	17350	LAR	13495	LAR	33736	LAR	29881
KON	15641	KON	12165	KON	30413	KON	26937
ASO	15549	ASO	12093	ASO	30234	ASO	26778
MOL	14855	MOL	11554	MOL	28885	MOL	25584
HAR	14832	HAR	11536	HAR	28840	HAR	25544
GJO	14494	GJO	11273	GJO	28182	GJO	24961
LIL	14487	LIL	11268	LIL	28170	LIL	24950
HOR	14434	HOR	11226	HOR	28066	HOR	24858
JES	14090	JES	10959	JES	27398	JES	24266
SKI	13759	SKI	10702	SKI	26754	SKI	23697
MOI	13304	MOI	10347	MOI	25869	MOI	22912
KRU	12863	KRU	10005	KRU	25012	KRU	22153
KOR	12658	KOR	9845	KOR	24612	KOR	21799
TRD	12232	TRD	9514	TRD	23785	TRD	21067
HON	12006	HON	9338	HON	23345	HON	20677
ALT	10800	ALT	8400	ALT	21000	ALT	18600
ELV	10642	ELV	8277	ELV	20692	ELV	18327
STJ	10364	STJ	8061	STJ	20153	STJ	17849
ASK	10199	ASK	7932	ASK	19831	ASK	17565
NAR	9959	NAR	7746	NAR	19366	NAR	17152
LEI	9944	LEI	7734	LEI	19335	LEI	17126
OSO	9793	OSO	7616	OSO	19041	OSO	16865
RAH	9506	RAH	7394	RAH	18484	RAH	16372
DRO	9428	DRO	7333	DRO	18332	DRO	16237
GRI	9365	GRI	7284	GRI	18210	GRI	16129
VEN	9234	VEN	7182	VEN	17956	VEN	15904
NES	9205	NES	7159	NES	17898	NES	15853
STE	9141	STE	7109	STE	17774	STE	15742
BRY	8590	BRY	6681	BRY	16702	BRY	14793
KOV	8471	KOV	6589	KOV	16472	KOV	14589
KOP	8138	KOP	6330	KOP	15825	KOP	14016
KNA	8097	KNA	6298	KNA	15744	KNA	13944
EGE	8048	EGE	6260	EGE	15649	EGE	13861
ALG	7979	ALG	6206	ALG	15515	ALG	13742
LOM	7884	LOM	6132	LOM	15330	LOM	13578
MAN	7679	MAN	5973	MAN	14932	MAN	13226
AAS	7650	AAS	5950	AAS	14876	AAS	13176
BRU	7504	BRU	5836	BRU	14591	BRU	12924
FOR	7278	FOR	5661	FOR	14152	FOR	12535
LEV	7274	LEV	5657	LEV	14144	LEV	12527
KOU	7261	KOU	5647	KOU	14118	KOU	12504
	2193702		1706213		4265531		3778042

Figure 3.4.5.6. Excel file

AM	AN	AO	AP
ell/YX	Required annual demand in 1000 litre for Circle K/Esso	o/YX	Required annual demand in 1000 litre for Shell/Esso/Y
OSL	1076523	OSL	1076523
BER	271458	BER	271458
STA	237603	STA	237603
TRH	136785	TRH	136785
FRE	119976	FRE	119976
DRA	113965	DRA	113965
POR	98470	POR	98470
KRI	67639	KRI	67639
ALE	56232	ALE	56232
TON	55350	TON	55350
MOS	49771	MOS	49771
HAU	47559	HAU	47559
SAN	46849	SAN	46849
ARE	45948	ARE	45948
BOD	44053	BOD	44053
TRO	42734	TRO	42734
HAM	29510	HAM	29510
HAL	27146	HAL	27146
LAR	26025	LAR	26025
KON	23461	KON	23461
ASO	23323	ASO	23323
MOL	22283	MOL	22283
HAR	22248	HAR	22248
GJO	21740	GJO	21740
LIL	21731	LIL	21731
HOR	21651	HOR	21651
JES	21135	JES	21135
SKI	20639	SKI	20639
MOI	19956	MOI	19956
KRU	19295	KRU	19295
KOR	18986	KOR	18986
TRD	18349	TRD	18349
HON	18009	HON	18009
ALT	16200	ALT	16200
ELV	15962	ELV	15962
STJ	15546	STJ	15546
ASK	15298	ASK	15298
NAR	14939	NAR	14939
LEI	14916	LEI	14916
OSO	14689	OSO	14689
RAH	14259	RAH	14259
DRO	14142	DRO	14142
GRI	14048	GRI	14048
VEN	13852	VEN	13852
NES	13807	NES	13807
STE	13711	STE	13711
BRY	12884	BRY	12884
KOV	12707	KOV	12707
KOP	12207	KOP	12207
KNA	12145	KNA	12145
EGE	12072	EGE	12072
ALG	11969	ALG	11969
LOM	11826	LOM	11826
MAN	11519	MAN	11519
AAS	11476	AAS	11476
BRU	11256	BRU	11256
FOR	10917	FOR	10917
LEV	10911	LEV	10911
KOU	10891	KOU	10891
	3290553		3290553

3.4.6 Cost for transportation

I have calculated the cost parameter “c” manually, from a source from internet, own knowledge and statements from people in the industry. This is hence an estimated parameter, but I believe the parameters value is not far from the true:

Cost for 1 litre diesel on average in Norway October 2020 = 13.24 NOK (GlobalPetrolPrices, 2020)

From own knowledge and statements from people in the driver industry a gas truck’s tank contains when full tank typical 40000 litre and a truck driving with full tank of that amount consumes at average approximately 0.4 litre diesel per km driving.

Tank volum at truck: 40000 litre

To get the value for per 1000 litre: 40000 / 1000 = 40

0.4 / 40 = 0.01

This gives that a truck’s consumption of diesel is estimated to 0.01 litre diesel per 1000 litre km, given that the truck is driven with full tank of 40000 litre on the truck.

0.01 x 13.24 = 0.1324

This gives that cost for consume of diesel is 0.1324 NOK per 1000 litre km

Based on rest time for the driver and estimated salary; salary: 200 NOK/hour

Based on own observation on average speed: 65 km/hour

200 / 65 = 3.077

This gives salary 3.077 NOK per km, given the truck are driven with full tank of 40000 litre

3.077 / 40 = 0.077

This gives salary 0.077 NOK per 1000 litre km

Estimated maintenance cost for truck: 0.1 per 1000 litre km

0.1324 + 0.077 + 0.1 = 0.3094

This gives transportation cost: 0.31 NOK per 1000 litre km

3.4.7 Sets and parameters

In the data files I will keep all the data for the analyses. The data files are organized in sets and parameters, and gives names to the sets and values to the parameters.

“set I” gives names (initials) to all the 20 depots.

“set J” gives names to all the 59 cities/places.

Param “k” gives values to all “I” to “J”, which is all the 1180 distances between depots and the gas stations cities/places.

Param “s” gives values to all “I”, which is max supply fuel in 1000 litres for each depot, which is a restriction.

Param “d” gives values to all “J”, which is required demand fuel in 1000 litre for each city/place, which is a restriction.

Param “c” gives a value to the cost constant “c” for the cost of transporting fuel per 1000 litre KM in NOK, which is a part of the minimizing formula.

3.4.8 Data file

The data files are containing a lot of data, particularly regarding distances, each of the data files contains 1180 distances. I will here show several screen shots were all these screen shots are parts of the same data file (the data file from the analysis where we have full collaboration included Equinor’s depot at Mongstad, with “equal” depot capacities).

Figure 3.4.8.1. Data file

```
*Data.dat    Model.mod    Run.run    Text.txt

# I are set of depots
# J are set of cities/places

set I:= CIALT, CIHAR, CITRH, CIFOR, CIKRI, CIOSL, SHTAN, SHVES, SHSKJ, SHKIR, SHBAL, SHLAR, SHLIL, SHSJU, ESTRH, ESBER, ESSLA, ESFRE, YXMOI, YXSTA, EQMON; # dep
set J:= OSL, BER, STA, TRH, FRE, DRA, POR, KRI, ALE, TON, MOS, HAU, SAN, ARE, BOD, TRO, HAM, HAL, LAR, KON, ASO MOL, HAR  GJO, LIL, HOR, JES, SKI, MOI, KRU, KOR

param k :
/*      The following initials indicate depots owned by Circle K at respective places:
        CIALT: Alta
        CIHAR: Harstad
        CITRH: Trondheim
        CIFOR: Førde
        CIKRI: Kristiansand
        CIOSL: Oslo

        The following initials indicate depots owned by Shell/ST1 at respective places:
        SHTAN: Tananger
        SHVES: Vestervika
        SHSKJ: Skjelnan
        SHKIR: Kirkenes
        SHBAL: Balsfjord
        SHLAR: Larsgården
        SHLIL: Lillesund
        SHSJU: Sjursøya

        The following initials indicate depots owned by Esso at respective places:
        ESTRH: Trondheim
        ESBER: Bergen
        ESSLA: Slagen
        ESFRE: Fredrikstad

        The following initials indicate depots owned by UnoX/YX at respective places:
        YXMOI: Mo i Rana
        YXSTA: Stavanger

        The following initial indicate depot owned by Equinor at respective place:
        EQMON: Mongstad (Equinor do not own/operate gas stations)

        The following initials indicate all cities/places for gas stations(for all 4 companies) of minimum 10000 citizen in Norway:
        OSL: Oslo
        BER: Bergen
        STA: Stavanger
```

Figure 3.4.8.2. Data file

```
Data.dat

ION; # depots
, KRU, KOR, TRD HON, ALT, ELV, STJ, ASK, NAR, LEI, OSO, RAH, DRO, GRI, VEN, NES, STE, BRY, KOV, KOP, KNA, EGE, ALG LOM, MAN, AAS, BRU, FOR, LEV, KOU; # gas stations
```

Figure 3.4.8.3. Data file

```
*Data.dat    Model.mod    Run.run    Text.txt

The following initials indicate depots owned by UnoX/YX at respective places:
YXMOI: Mo i Rana
YXSTA: Stavanger

The following initial indicate depot owned by Equinor at respective place:
EQMON: Mongstad (Equinor do not own/operate gas stations)

The following initials indicate all cities/places for gas stations(for all 4 companies) of minimum 10000 citizen in Norway:
OSL: Oslo
BER: Bergen
STA: Stavanger
TRH: Trondheim
FRE: Fredrikstad
DRA: Drammen
POR: Porsgrunn
KRI: Kristiansand
ALE: Ålesund
TON: Tønsberg
MOS: Moss
HAU: Haugesund
SAN: Sandefjord
ARE: Arendal
BOD: Bodø
TRO: Tromsø
HAM: Hamar
HAL: Halden
LAR: Larvik
KON: Kongsberg
ASO: Askøy
MOL: Molde
HAR: Harstad
GJO: Gjøvik
LIL: Lillehammer
HOR: Horten
JES: Jessheim
SKI: Ski
MOI: Mo i Rana
KRU: Kristiansund
KOR: Korsvik
TRD: Tromsdalen
HON: Hønefoss
```

Figure 3.4.8.4. Data file

```
*Data.dat    Model.mod    Run.run    Text.txt

HAL: Halden
LAR: Larvik
KON: Kongsberg
ASO: Askøy
MOL: Molde
HAR: Harstad
GJO: Gjøvik
LIL: Lillehammer
HOR: Horten
JES: Jessheim
SKI: Ski
MOI: Mo i Rana
KRU: Kristiansund
KOR: Korsvik
TRD: Tromsdalen
HON: Hønefoss
ALT: Alta
ELV: Elverum
STJ: Stjørdalshalsen
ASK: Askim
NAR: Narvik
LEI: Leirvik
OSO: Osøyro
RAH: Råholt
DRO: Drøbak
GRI: Grimstad
VEN: Vennesla
NES: Nesoddtangen
STE: Steinkjer
BRY: Bryne
KOV: Kongsvinger
KOP: Kopervik
KNA: Knarrvika
EGE: Egersund
ALG: Ålgård
LOM: Lommedalen
MAN: Mandal
AAS: Ås
BRU: Brummunddal
FOR: Førde
LEV: Levanger
KOU: Konnerud
```

Figure 3.4.8.5. Data file

*Data.dat | Model.mod | Run.run | Text.txt

```
        The data in the following table are distances in km between depots and cities/places of gas stations:
        */
        OSL   BER   STA   TRH   FRE   DRA   POR   KRI   ALE   TON   MOS   HAU   SAN   ARE   BOD   TRO   HAM   HAL   LAR
CIALT   1733  2094  2286  1390  1821  1777  1891  2053  1685  1837  1789  2177  1856  1993  770   380   1623  1838  1869
CIHAR   1397  1602  1950  910   1485  1441  1555  1716  1205  1501  1453  1724  1520  1657  313   300   1287  1511  1532
CITRH   491   627   933   1     579   535   650   810   299   595   547   745   614   751   701   1128  381   605   627
CIFOR   419   174   380   529   508   417   534   696   247   480   476   309   499   636   1226  1653  417   534   512
CIKRI   324   469   235   822   293   285   176   1     870   230   255   314   209   71    1520  2064  452   319   193
CIOSL   7     467   447   497   88    47    162   323   550   107   56    447   126   264   1195  1747  132   114   139
SHTAN   445   212   13    1048  524   402   406   237   630   461   486   85    440   445   1745  2297  683   550   424
SHVES   1192  1397  1744  704   1279  1235  1350  1511  999   1295  1247  1518  1314  1452  5     534   1081  1305  1327
SHSKJ   1746  1826  2299  1133  1834  1790  1905  2065  1429  1850  1802  1947  1869  2006  537   10    1636  1860  1882
SHKIR   1884  2191  2275  1519  1918  1874  1989  2150  1815  1934  1886  2275  1953  2090  1214  787   1720  1880  1965
SHBAL   1543  1748  2095  1055  1783  1738  1852  2015  1351  1798  1751  1870  1816  1954  459   99    1432  1809  1831
SHLAR   525   417   623   295   628   585   699   860   6     611   596   562   630   801   993   1420  419   654   676
SHLIL   445   148   81    822   510   402   367   312   565   430   478   11    410   377   1519  1946  573   536   393
SHSJU   7     467   447   497   88    47    162   323   550   107   56    447   126   264   1195  1747  132   114   139
ESTRH   491   627   933   1     579   535   650   810   299   595   547   745   614   751   701   1128  381   605   627
ESBER   464   1     210   698   553   443   402   468   422   484   521   139   492   462   1395  1822  477   579   428
ESSLA   103   483   468   595   61    63    74    234   648   12    26    436   38    175   1293  1844  230   87    50
ESFRE   92    555   521   579   1     108   127   288   632   69    39    508   91    229   1277  1828  214   37    104
YXMOI   954   1159  1507  467   1042  998   1112  1273  762   1058  1010  1281  1077  1214  235   662   844   1068  1089
YXSTA   445   212   13    1048  524   402   406   237   630   461   486   85    440   445   1745  2297  683   550   424
EQMON   498   64    270   657   586   477   436   502   375   518   554   199   479   496   1354  1781  510   612   462
;

param s :=
# max annual supply of fuel from depots in 1000 litres
CIALT   350000
CIHAR   350000
CITRH   350000
CIFOR   350000
CIKRI   350000
CIOSL   350000
SHTAN   350000
SHVES   350000
SHSKJ   350000
SHKIR   350000
SHBAL   350000
SHLAR   350000
SHLIL   350000
```

Figure 3.4.8.6. Data file

Data.dat

```
KON   ASO   MOL   HAR   GJO   LIL   HOR   JES   SKI   MOI   KRU   KOR   TRD   HON   ALT   ELV   STJ   ASK   NAR   LEI
1818  2119  1605  538   1668  1680  1828  1692  1755  893   1583  2051  379   1743  1     1593  1356  1811  470   2204
1482  1627  1125  1     1283  1241  1492  1356  1418  437   1103  1714  298   1407  538   1263  876   1447  100   1750
576   651   220   909   377   336   586   450   512   474   198   809   1127  470   1389  357   34    541   897   682
424   199   275   1434  368   381   471   436   445   999   345   694   1652  363   1956  444   559   470   1422  254
282   493   821   1728  432   508   244   367   350   1292  895   8     2062  342   2055  467   853   301   1886  359
89    492   502   1403  132   189   99    48    25    967   575   321   1745  61    1736  147   528   50    1390  474
366   237   658   1953  537   581   475   598   471   1518  863   244   2296  440   2286  698   1078  496   1943  130
1277  1421  920   316   1077  1036  1287  1151  1213  231   898   1509  533   1170  772   1057  671   1241  303   1545
1831  1851  1349  305   1506  1465  1841  1705  1767  660   1327  2064  7     1756  299   1606  1100  1796  237   1974
1916  2216  1735  917   1765  1777  1926  1789  1851  1191  1713  2148  786   1840  462   1689  1486  1853  832   2302
1628  1773  1271  227   1429  1387  1789  1502  1717  582   1249  2011  88    1553  337   1555  1022  1745  159   1896
578   442   77    1201  403   361   602   500   562   765   148   858   1418  495   1709  446   326   591   1188  497
366   173   593   1728  477   490   431   488   471   1292  664   319   2179  440   2177  588   852   496   1715  66
89    492   502   1403  132   189   99    48    25    967   575   321   1745  61    1736  147   528   50    1390  474
576   651   220   909   377   336   586   450   512   474   198   809   1127  470   1389  357   34    541   897   682
403   26    449   1603  428   440   475   481   489   1168  520   475   1826  408   2092  504   728   515   1590  84
82    508   599   1501  211   287   15    146   65    1065  673   232   1843  121   1834  245   626   71    1488  463
150   580   583   1485  214   271   52    130   71    1049  657   286   1827  149   1818  229   610   55    1472  535
1039  1184  682   443   840   798   1049  913   975   7     660   1271  660   964   900   820   433   1004  430   1307
366   237   658   1953  537   581   475   598   471   1518  863   244   2296  440   2286  698   1078  496   1943  130
437   89    403   1563  461   474   509   515   523   1127  474   508   1780  442   2126  538   687   548   1550  144
```

Figure 3.4.8.7. Data file

```
Data.dat

OSO   RAH   DRO   GRI   VEN   NES   STE   BRY   KOV   KOP   KNA   EGE   ALG   LOM   MAN   AAS   BRU   FOR   LEV   KCU :=
2094  1676  1767  2010  2065  1776  1270  2271  1655  2191  2105  2232  2258  1753  2096  1762  1636  1909  1311  1784
1677  1340  1430  1673  1729  1440  790   1935  1355  1737  1613  1895  1922  1417  1760  1425  1299  1429  831   1448
627   434   524   767   823   534   120   922   449   759   638   990   905   511   854   520   394   523   81    542
201   434   453   653   603   463   645   408   493   326   185   452   406   405   568   448   403   1     606   419
414   383   326   52    18    342   938   221   418   308   480   181   208   319   46    286   464   700   899   285
 467   64    33    280   335   43    614   436   99    461   478   502   419   29    367   28    145   423   574   55
 185   614   445   283   245   461   1164  31    649   78    223   77    31    440   194   453   695   383   1125  402
 1423  1134  1225  1468  1523  1234  585   1730  1150  1532  1408  1690  1717  1211  1555  1220  1094  1223  625   1243
 1852  1689  1779  2022  2078  1789  1014  2284  1694  1961  1848  2245  2271  1766  2109  1775  1649  1652  1054  1797
 2191  1773  1863  2107  2163  1873  1402  2264  1751  2288  1932  2329  2247  1856  2194  1859  1732  2039  1443  1882
 1774  1485  1729  1970  2026  1738  936   2081  1646  1883  1759  2041  2068  1563  2058  1724  1445  1575  977   1594
 443   483   574   817   872   583   411   651   515   569   428   694   649   536   904   569   405   243   372   558
 120   505   445   359   321   461   938   109   539   15    161   152   107   440   269   453   512   319   899   402
 467   64    33    280   335   43    614   436   99    461   478   502   419   29    367   28    145   423   574   55
 627   434   524   767   823   534   120   922   449   759   638   990   905   511   854   520   394   523   81    542
 30    479   486   466   452   508   814   238   546   156   13    281   236   450   398   493   462   175   775   443
 437   162   104   191   247   120   711   453   196   450   494   413   440   97    278   57    243   479   672   64
 555   146   69    245   300   87    695   507   180   522   566   467   494   117   332   63    227   511   656   114
 1186  897   987   1230  1286  997   347   1492  912   1294  1170  1452  1479  974   1317  982   856   986   388   1005
 185   614   445   283   245   461   1164  31    649   78    223   77    31    440   194   453   695   383   1125  402
 90    512   532   500   486   541   773   298   572   216   75    341   296   484   458   527   496   129   734   477
```

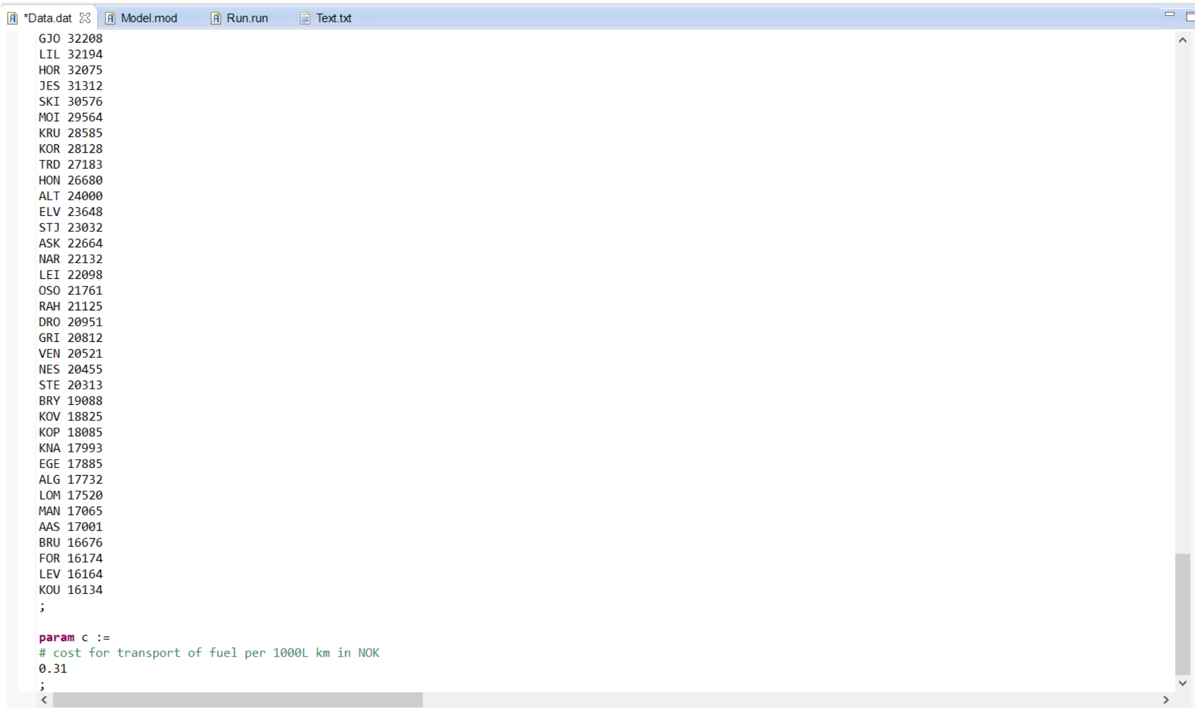
Figure 3.4.8.8. Data file

```
*Data.dat   Model.mod   Run.run   Text.txt

ESTRH   350000
ESBER   350000
ESSLA   350000
ESFRE   350000
YXMOI   350000
YXSTA   350000
EQMON   350000
;

param d :=
# required annual total demand of fuel at the gas stations cities/places in 1000 litres
OSL 1594850
BER 402160
STA 352004
TRH 291534
FRE 177742
DRA 168838
POR 145881
KRI 100206
ALE 83307
TON 82000
MOS 73734
HAU 70457
SAN 69406
ARE 68072
BOD 65264
TRO 63310
HAM 43718
HAL 40216
LAR 38556
KON 34758
ASO 34553
MOL 33012
HAR 32960
GJO 32208
LIL 32194
HOR 32075
JES 31312
SKI 30576
MOI 29564
KRU 28585
KOR 28128
```

Figure 3.4.8.9. Data file




```
GJO 32208
LIL 32194
HOR 32075
JES 31312
SKI 30576
MOI 29564
KRU 28585
KOR 28128
TRD 27183
HON 26680
ALT 24000
ELV 23648
STJ 23032
ASK 22664
NAR 22132
LEI 22098
OSO 21761
RAH 21125
DRO 20951
GRI 20812
VEN 20521
NES 20455
STE 20313
BRY 19088
KOV 18825
KOP 18085
KNA 17993
EGE 17885
ALG 17732
LOM 17520
MAN 17065
AAS 17001
BRU 16676
FOR 16174
LEV 16164
KOU 16134
;

param c :=
# cost for transport of fuel per 1000L km in NOK
0.31
;
```

3.5 Run file

3.5.1 Comandoes

The run file here content several commands for the program.

It first tells the program to reset, so that no stored codes will disturb from previous analyses.

Then it tells the program what model file to use, which here is “Model.mod”.

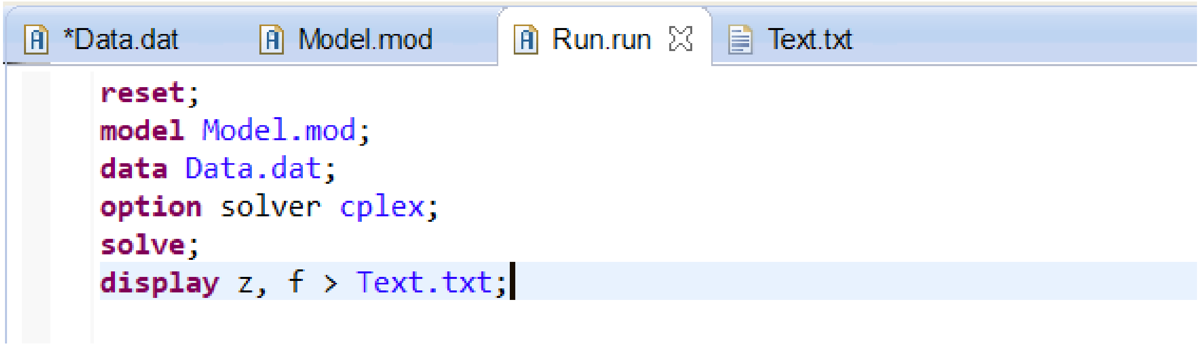
Then it tells the program what data file to use, which here is “Data.dat”.

Then it tells the program which solver to use, which here is cplex.

Then it tells the program what to display in which text file, which here is from the optimized solution to both display the minimized cost “z”, and “f” which is a table of all the quantities for all the corresponding routes “I” to “J”, to the text file “Text.txt”.

3.5.2 Run file

Figure 3.5.2.1. Run file



3.6 Text file

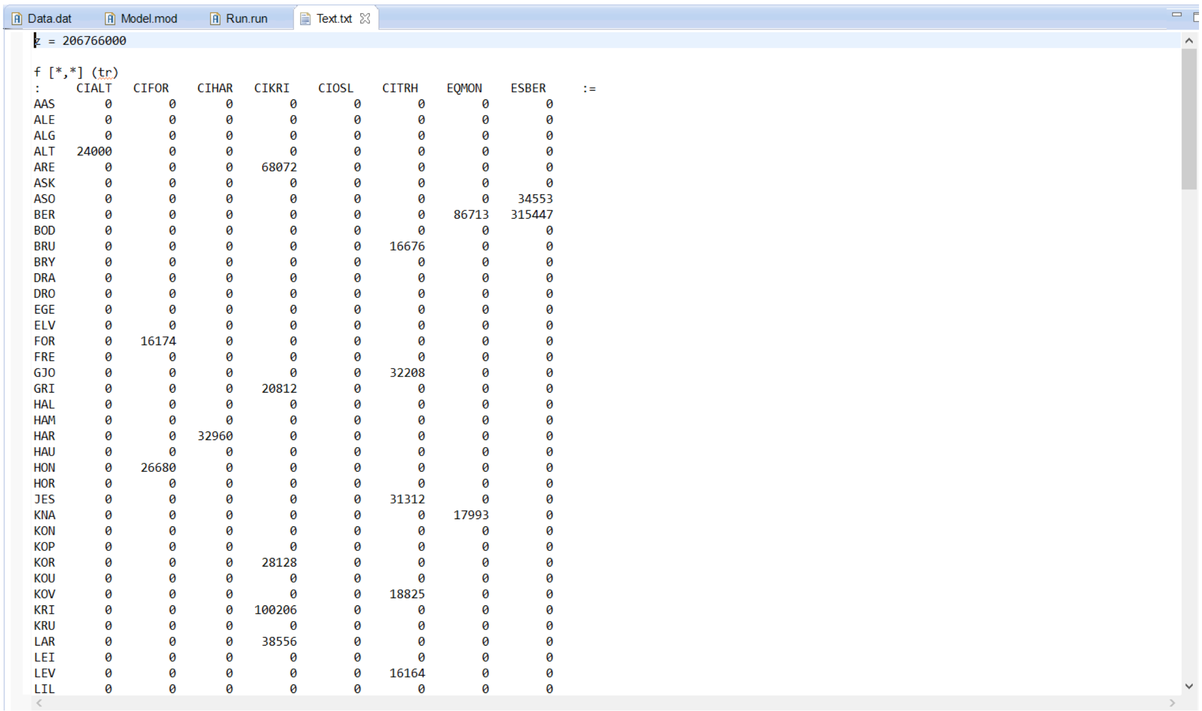
3.6.1 Explanation

The text file now from a optimized solution first shows the minimized annual cost “z” in NOK.

Further it shows “f”, which is all the amounts in 1000 litres for all corresponding routes “I” to “J” the optimized solution have chosen. This is the amount of fuel each depot should transport to what city/place in an annual basis which will give the lowest possible total cost, given the restrictions. This analyse apply to the analysis variant where all 4 companies collaborating including that they can transport fuel from Equinor’s depot at Mongstad.

3.6.2 Text file

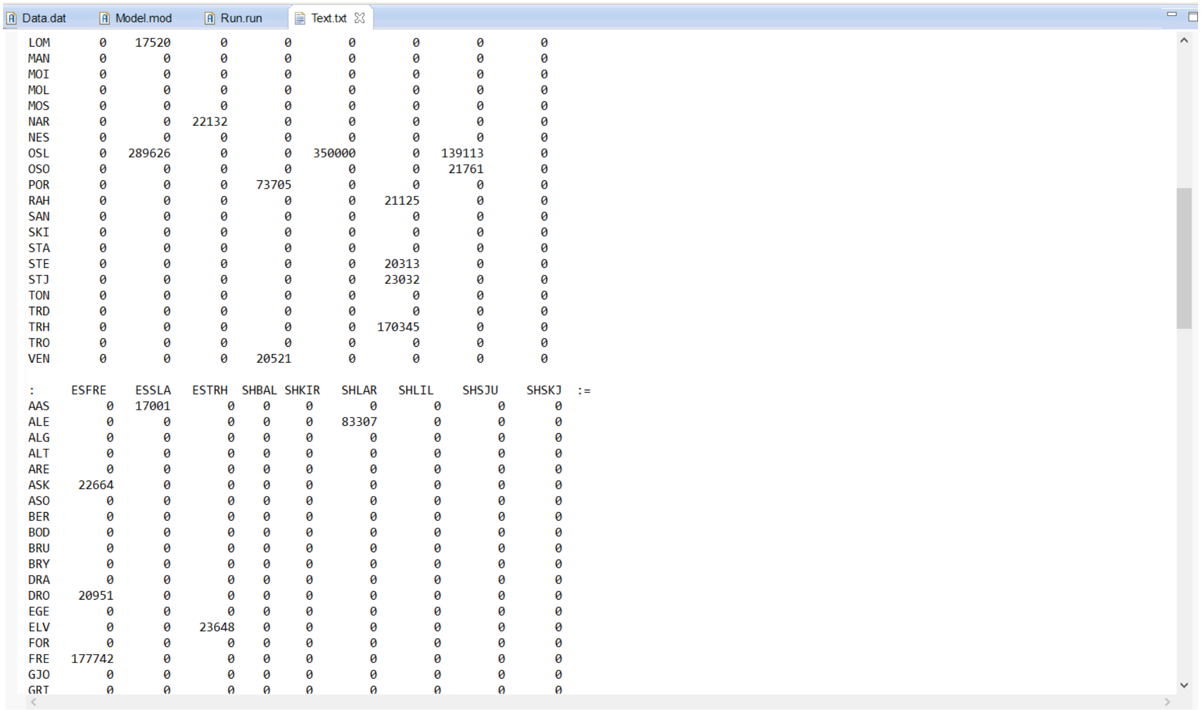
Figure 3.6.2.1. Text file



```
z = 206766000

f [*,*] (tr)
:     CIALT  CIFOR  CIHAR  CIKRI  CIOSL  CITRH  EQMON  ESBER   :=
AAS       0      0      0      0      0      0      0      0
ALE       0      0      0      0      0      0      0      0
ALG       0      0      0      0      0      0      0      0
ALT   24000      0      0      0      0      0      0      0
ARE       0      0      0  68072      0      0      0      0
ASK       0      0      0      0      0      0      0      0
ASO       0      0      0      0      0      0      0  34553
BER       0      0      0      0      0      0  86713 315447
BOD       0      0      0      0      0      0      0      0
BRU       0      0      0      0      0  16676      0      0
BRY       0      0      0      0      0      0      0      0
DRA       0      0      0      0      0      0      0      0
DRO       0      0      0      0      0      0      0      0
EGE       0      0      0      0      0      0      0      0
ELV       0      0      0      0      0      0      0      0
FOR       0  16174      0      0      0      0      0      0
FRE       0      0      0      0      0      0      0      0
GJO       0      0      0      0      0  32208      0      0
GRI       0      0      0  20812      0      0      0      0
HAL       0      0      0      0      0      0      0      0
HAM       0      0      0      0      0      0      0      0
HAR       0      0  32960      0      0      0      0      0
HAU       0      0      0      0      0      0      0      0
HON       0  26680      0      0      0      0      0      0
HOR       0      0      0      0      0      0      0      0
JES       0      0      0      0      0  31312      0      0
KNA       0      0      0      0      0      0  17993      0
KON       0      0      0      0      0      0      0      0
KOP       0      0      0      0      0      0      0      0
KOR       0      0      0  28128      0      0      0      0
KOU       0      0      0      0      0      0      0      0
KOV       0      0      0      0      0  18825      0      0
KRI       0      0      0 100206      0      0      0      0
KRU       0      0      0      0      0      0      0      0
LAR       0      0      0  38556      0      0      0      0
LEI       0      0      0      0      0      0      0      0
LEV       0      0      0      0      0  16164      0      0
LIL       0      0      0      0      0      0      0      0
```

Figure 3.6.2.2. Text file



```
LOM       0  17520      0      0      0      0      0      0
MAN       0      0      0      0      0      0      0      0
MOI       0      0      0      0      0      0      0      0
MOL       0      0      0      0      0      0      0      0
MOS       0      0      0      0      0      0      0      0
NAR       0      0  22132      0      0      0      0      0
NES       0      0      0      0      0      0      0      0
OSL       0 289626      0      0 350000      0 139113      0
OSO       0      0      0      0      0      0  21761      0
POR       0      0      0  73705      0      0      0      0
RAH       0      0      0      0      0  21125      0      0
SAN       0      0      0      0      0      0      0      0
SKI       0      0      0      0      0      0      0      0
STA       0      0      0      0      0      0      0      0
STE       0      0      0      0      0  20313      0      0
STJ       0      0      0      0      0  23032      0      0
TON       0      0      0      0      0      0      0      0
TRD       0      0      0      0      0      0      0      0
TRH       0      0      0      0      0 170345      0      0
TRO       0      0      0      0      0      0      0      0
VEN       0      0      0  20521      0      0      0      0

:    ESFRE  ESSLA  ESTRH SHBAL SHKIR  SHLAR  SHLIL  SHSJU  SHSKJ  :=
AAS       0  17001      0     0     0      0      0      0      0
ALE       0      0      0     0     0  83307      0      0      0
ALG       0      0      0     0     0      0      0      0      0
ALT       0      0      0     0     0      0      0      0      0
ARE       0      0      0     0     0      0      0      0      0
ASK   22664      0      0     0     0      0      0      0      0
ASO       0      0      0     0     0      0      0      0      0
BER       0      0      0     0     0      0      0      0      0
BOD       0      0      0     0     0      0      0      0      0
BRU       0      0      0     0     0      0      0      0      0
BRY       0      0      0     0     0      0      0      0      0
DRA       0      0      0     0     0      0      0      0      0
DRO   20951      0      0     0     0      0      0      0      0
EGE       0      0      0     0     0      0      0      0      0
ELV       0      0  23648     0     0      0      0      0      0
FOR       0      0      0     0     0      0      0      0      0
FRE  177742      0      0     0     0      0      0      0      0
GJO       0      0      0     0     0      0      0      0      0
GRI       0      0      0     0     0      0      0      0      0
```

Figure 3.6.2.3. Text file

Data.dat Model.mod Run.run Text.txt

GRI	0	0	0	0	0	0	0	0	0
HAL	40216	0	0	0	0	0	0	0	0
HAM	0	0	43718	0	0	0	0	0	0
HAR	0	0	0	0	0	0	0	0	0
HAU	0	0	0	0	0	0	70457	0	0
HON	0	0	0	0	0	0	0	0	0
HOR	0	32075	0	0	0	0	0	0	0
JES	0	0	0	0	0	0	0	0	0
KNA	0	0	0	0	0	0	0	0	0
KON	0	0	0	0	0	0	0	0	0
KOP	0	0	0	0	0	0	18085	0	0
KOR	0	0	0	0	0	0	0	0	0
KOU	0	0	0	0	0	0	0	0	0
KOV	0	0	0	0	0	0	0	0	0
KRI	0	0	0	0	0	0	0	0	0
KRU	0	0	0	0	0	28585	0	0	0
LAR	0	0	0	0	0	0	0	0	0
LEI	0	0	0	0	0	0	22098	0	0
LEV	0	0	0	0	0	0	0	0	0
LIL	0	0	32194	0	0	0	0	0	0
LOM	0	0	0	0	0	0	0	0	0
MAN	0	0	0	0	0	0	0	0	0
MOI	0	0	0	0	0	0	0	0	0
MOL	0	0	0	0	0	33012	0	0	0
MOS	0	73734	0	0	0	0	0	0	0
NAR	0	0	0	0	0	0	0	0	0
NES	20455	0	0	0	0	0	0	0	0
OSL	67972	45208	129251	0	0	0	167184	350000	0
OSO	0	0	0	0	0	0	0	0	0
POR	0	0	0	0	0	0	72176	0	0
RAH	0	0	0	0	0	0	0	0	0
SAN	0	69406	0	0	0	0	0	0	0
SKI	0	30576	0	0	0	0	0	0	0
STA	0	0	0	0	0	0	0	0	0
STE	0	0	0	0	0	0	0	0	0
STJ	0	0	0	0	0	0	0	0	0
TON	0	82000	0	0	0	0	0	0	0
TRD	0	0	0	0	0	0	0	0	27183
TRH	0	0	121189	0	0	0	0	0	0
TRO	0	0	0	0	0	0	0	0	63310
VEN	0	0	0	0	0	0	0	0	0

Figure 3.6.2.4. Text file

Data.dat Model.mod Run.run Text.txt

:	SHTAN	SHVES	YXMOI	YXSTA
AAS	0	0	0	0
ALE	0	0	0	0
ALG	17732	0	0	0
ALT	0	0	0	0
ARE	0	0	0	0
ASK	0	0	0	0
ASO	0	0	0	0
BER	0	0	0	0
BOD	0	65264	0	0
BRU	0	0	0	0
BRY	19088	0	0	0
DRA	0	0	0	168838
DRO	0	0	0	0
EGE	0	0	0	17885
ELV	0	0	0	0
FOR	0	0	0	0
FRE	0	0	0	0
GJO	0	0	0	0
GRI	0	0	0	0
HAL	0	0	0	0
HAM	0	0	0	0
HAR	0	0	0	0
HAU	0	0	0	0
HON	0	0	0	0
HOR	0	0	0	0
JES	0	0	0	0
KNA	0	0	0	0
KON	0	0	0	34758
KOP	0	0	0	0
KOR	0	0	0	0
KOU	16134	0	0	0
KOV	0	0	0	0
KRI	0	0	0	0
KRU	0	0	0	0
LAR	0	0	0	0
LEI	0	0	0	0
LEV	0	0	0	0
LIL	0	0	0	0
LOM	0	0	0	0
MAN	17065	0	0	0
MOI	0	0	29564	0

Figure 3.6.2.5. Text file

HAM	0	0	0	0
HAR	0	0	0	0
HAU	0	0	0	0
HON	0	0	0	0
HOR	0	0	0	0
JES	0	0	0	0
KNA	0	0	0	0
KON	0	0	0	34758
KOP	0	0	0	0
KOR	0	0	0	0
KOU	16134	0	0	0
KOV	0	0	0	0
KRI	0	0	0	0
KRU	0	0	0	0
LAR	0	0	0	0
LEI	0	0	0	0
LEV	0	0	0	0
LIL	0	0	0	0
LOM	0	0	0	0
MAN	17065	0	0	0
MOI	0	0	29564	0
MOL	0	0	0	0
MOS	0	0	0	0
NAR	0	0	0	0
NES	0	0	0	0
OSL	0	0	0	56496
OSO	0	0	0	0
POR	0	0	0	0
RAH	0	0	0	0
SAN	0	0	0	0
SKI	0	0	0	0
STA	279981	0	0	72023
STE	0	0	0	0
STJ	0	0	0	0
TON	0	0	0	0
TRD	0	0	0	0
TRH	0	0	0	0
TRO	0	0	0	0
VEN	0	0	0	0

;

4. Changes in the AMPL files

4.1 Stand-alone analysis

In stand-alone all companies work for them self and no collaboration is done in the distribution of fuel. To analyze this in AMPL it require in total 4 unique data files, 4 unique run files, 4 unique text files, which all are different from the files at full collaboration analysis, and the model file which I keep unchanged as mentioned earlier.

4.1.1 Data file

Circle K

Changes here is in parameter s and in parameter d. Changes in d for changed capasities, do to the fact that this analysis apply only to the cost of Circle K's stand alone cost for their own needs for fuel distributed from their depots to their gas stations. This gives zero to the other companies s.

Parameter d is changed as well as Circle K only have 0.325 of the fuel market.

Below I show changes in the data file for the Circle K stand alone analysis.

I only give Circle K capacities, the other companies have now capacities zero.

Figure 4.1.1.1. Circle K stand-alone data file

DataCi.dat

```
param s :=
# max annual supply of fuel from depots in 1000 litres
CIALT   350000
CIHAR   350000
CITRH   350000
CIFOR   350000
CIKRI   350000
CIOSL   350000
SHTAN   0
SHVES   0
SHSKJ   0
SHKIR   0
SHBAL   0
SHLAR   0
SHLIL   0
SHSJU   0
ESTRH   0
ESBER   0
ESSLA   0
ESFRE   0
YXMOI   0
YXSTA   0
EQMON   0
;
```

I have now adjusted the demands for the places/cities to match the demands of Circle K's gas stations only, which is 0.325 of the market. This numbers I have calculated in the excel file Thesis Data.

Figure 4.1.1.2. Circle K stand-alone data file

DataCi.dat

```
param d :=
# required annual total demand of fuel at the gas stations cities/places in 1000 litres
OSL 518326
BER 130702
STA 114401
TRH 94748
FRE 57766
DRA 54872
POR 47411
KRI 32567
ALE 27075
TON 26650
MOS 23964
HAU 22899
SAN 22557
ARE 22123
BOD 21211
TRO 20576
HAM 14208
HAL 13070
LAR 12531
KON 11296
ASO 11230
MOL 10729
HAR 10712
GJO 10468
LIL 10463
HOR 10424
JES 10176
SKI 9937
MOI 9608
KRU 9290
KOR 9142
TRD 8835
HON 8671
ALT 7800
ELV 7686
STJ 7485
ASK 7366
NAR 7193
LEI 7182
OSO 7072
RAH 6866
DRO 6809
GRI 6764
VEN 6669
NES 6648
STE 6602
BRY 6204
KOV 6118
KOP 5878
KNA 5848
EGE 5813
ALG 5763
LOM 5694
MAN 5546
AAS 5525
BRU 5420
FOR 5256
LEV 5253
KOU 5244
;

param c :=
# cost for transport of fuel per 1000L km in NOK
0.31
;
```

Further I have done equivalently prosedures in the analyzes of the other companies.

Shell

Figure 4.1.1.3. Shell stand-alone data file

```
DataSh.dat
# max annual supply of fuel from depots in 1000 litres
CIALT   0
CIHAR   0
CITRH   0
CIFOR   0
CIKRI   0
CIOSL   0
SHTAN   350000
SHVES   350000
SHSKJ   350000
SHKIR   350000
SHBAL   350000
SHLAR   350000
SHLIL   350000
SHSJU   350000
ESTRH   0
ESBER   0
ESSLA   0
ESFRE   0
YXMOI   0
YXSTA   0
EQMON   0
;
```

Figure 4.1.1.4. Shell stand-alone data file

```
DataSh.dat
param d :=
# required annual total demand of fuel at the gas stations cities/places in 1000 litres
OSL 518326
BER 130702
STA 114401
TRH 94748
FRE 57766
DRA 54872
POR 47411
KRI 32567
ALE 27075
TON 26650
MOS 23964
HAU 22899
SAN 22557
ARE 22123
BOD 21211
TRO 20576
HAM 14208
HAL 13070
LAR 12531
KON 11296
ASO 11230
MOL 10729
HAR 10712
GJO 10468
LIL 10463
HOR 10424
JES 10176
SKI 9937
MOI 9608
KRU 9290
KOR 9142
TRD 8835
HON 8671
ALT 7800
ELV 7686
STJ 7485
ASK 7366
NAR 7193
LEI 7182
```

Figure 4.1.1.5. Shell stand-alone data file

```
OSO 7072
RAH 6866
DRO 6809
GRI 6764
VEN 6669
NES 6648
STE 6602
BRY 6204
KOV 6118
KOP 5878
KNA 5848
EGE 5813
ALG 5763
LOM 5694
MAN 5546
AAS 5525
BRU 5420
FOR 5256
LEV 5253
KOU 5244
;

param c :=
# cost for transport of fuel per 1000L km in NOK
0.31
;
```

Esso

Figure 4.1.1.6. Esso stand-alone data file

```
DataEs.dat

param s :=
# max annual supply of fuel from depots in 1000 litres
CIALT   0
CIHAR   0
CITRH   0
CIFOR   0
CIKRI   0
CIOSL   0
SHTAN   0
SHVES   0
SHSKJ   0
SHKIR   0
SHBAL   0
SHLAR   0
SHLIL   0
SHSJU   0
ESTRH   350000
ESBER   350000
ESSLA   350000
ESFRE   350000
YXMOI   0
YXSTA   0
EQMON   0
;
```

Figure 4.1.1.7. Esso stand-alone data file

DataEs.dat

```
param d :=
# required annual total demand of fuel at the gas stations cities/places in 1000 litres
OSL 358841
BER 90486
STA 79201
TRH 65595
FRE 39992
DRA 37988
POR 32823
KRI 22546
ALE 18744
TON 18450
MOS 16590
HAU 15853
SAN 15616
ARE 15316
BOD 14684
TRO 14245
HAM 9837
HAL 9049
LAR 8675
KON 7820
ASO 7774
MOL 7428
HAR 7416
GJO 7247
LIL 7244
HOR 7217
JES 7045
SKI 6880
MOI 6652
KRU 6432
KOR 6329
TRD 6116
HON 6003
ALT 5400
ELV 5321
STJ 5182
ASK 5099
NAR 4980
LEI 4972
```

Figure 4.1.1.8. Esso stand-alone data file

```
OSO 4896
RAH 4753
DRO 4714
GRI 4683
VEN 4617
NES 4602
STE 4570
BRY 4295
KOV 4236
KOP 4069
KNA 4048
EGE 4024
ALG 3990
LOM 3942
MAN 3840
AAS 3825
BRU 3752
FOR 3639
LEV 3637
KOU 3630
;

param c :=
# cost for transport of fuel per 1000L km in NOK
0.31
;
```

YX

Figure 4.1.1.9. YX stand-alone data file

DataYX.dat

```
param s :=
# max annual supply of fuel from depots in 1000 litres
CIALT   0
CIHAR   0
CITRH   0
CIFOR   0
CIKRI   0
CIOSL   0
SHTAN   0
SHVES   0
SHSKJ   0
SHKIR   0
SHBAL   0
SHLAR   0
SHLIL   0
SHSJU   0
ESTRH   0
ESBER   0
ESSLA   0
ESFRE   0
YXMOI   350000
YXSTA   350000
EQMON   0
;
```

Figure 4.1.1.10. YX stand-alone data file

DataYX.dat

```
param d :=
# required annual total demand of fuel at the gas stations cities/places in 1000 litres
OSL 199356
BER 50270
STA 44001
TRH 36442
FRE 22218
DRA 21105
POR 18235
KRI 12526
ALE 10413
TON 10250
MOS 9217
HAU 8807
SAN 8676
ARE 8509
BOD 8158
TRO 7914
HAM 5465
HAL 5027
LAR 4819
KON 4345
ASO 4319
MOL 4126
HAR 4120
GJO 4026
LIL 4024
HOR 4009
JES 3914
SKI 3822
MOI 3696
KRU 3573
KOR 3516
TRD 3398
HON 3335
ALT 3000
ELV 2956
STJ 2879
ASK 2833
NAR 2767
LEI 2762
OSO 2720
```

Figure 4.1.1.11. YX stand-alone data file

```
OSO 2720
RAH 2641
DRO 2619
GRI 2601
VEN 2565
NES 2557
STE 2539
BRY 2386
KOV 2353
KOP 2261
KNA 2249
EGE 2236
ALG 2216
LOM 2190
MAN 2133
AAS 2125
BRU 2084
FOR 2022
LEV 2021
KOU 2017
;

param c :=
# cost for transport of fuel per 1000L km in NOK
0.31
;
```

4.1.2 Run file

Circle K

Figure 4.1.2.1. Circle K stand-alone run file

```
RunCi.run
reset;
model Model.mod;
data DataCi.dat;
option solver cplex;
solve;
display z, f > TextCi.txt;
```

Shell

Figure 4.1.2.2. Shell stand-alone run file

```
RunSh.run
reset;
model Model.mod;
data DataSh.dat;
option solver cplex;
solve;
display z, f > TextSh.txt;
```

Esso

Figure 4.1.2.3. Esso stand-alone run file

```
RunEs.run
reset;
model Model.mod;
data DataEs.dat;
option solver cplex;
solve;
display z, f > TextEs.txt;
```

YX

Figure 4.1.2.4. YX stand-alone run file

```
RunYX.run
reset;
model Model.mod;
data DataYX.dat;
option solver cplex;
solve;
display z, f > TextYX.txt;
```

4.1.3 Text file results

Circle K

Here we can see that only Circle K is distributing fuel to their gas stations at their demands from their depots within their capacities.

Figure 4.1.3.1. Circle K stand-alone text file

```
TextCi.txt
z = 131161000

f [*,*] (tr)
# $8 = ESBER
# $9 = ESFRE
# $10 = ESSLA
# $11 = ESTRH
# $12 = SHBAL
:     CIALT   CIFOR    CIHAR   CIKRI   CIOSL    CITRH  EQMON  $8   $9 $10 $11 $12 :=
AAS      0        0        0       0    5525        0     0    0    0   0   0   0
ALE      0        0        0       0       0    27075     0    0    0   0   0   0
ALG      0        0        0    5763       0        0     0    0    0   0   0   0
ALT   7800        0        0       0       0        0     0    0    0   0   0   0
ARE      0        0        0   22123       0        0     0    0    0   0   0   0
ASK      0        0        0       0    7366        0     0    0    0   0   0   0
ASO      0    11230        0       0       0        0     0    0    0   0   0   0
BER      0   130702        0       0       0        0     0    0    0   0   0   0
BOD      0        0    21211       0       0        0     0    0    0   0   0   0
BRU      0        0     5420       0       0        0     0    0    0   0   0   0
BRY      0        0        0    6204       0        0     0    0    0   0   0   0
DRA      0    54872        0       0       0        0     0    0    0   0   0   0
DRO      0        0        0       0    6809        0     0    0    0   0   0   0
EGE      0        0        0    5813       0        0     0    0    0   0   0   0
ELV      0        0        0       0       0     7686     0    0    0   0   0   0
FOR      0     5256        0       0       0        0     0    0    0   0   0   0
FRE      0        0        0   57766       0        0     0    0    0   0   0   0
GJO      0        0        0       0       0    10468     0    0    0   0   0   0
GRI      0        0        0    6764       0        0     0    0    0   0   0   0
HAL      0        0        0   13070       0        0     0    0    0   0   0   0
HAM      0        0        0       0       0    14208     0    0    0   0   0   0
HAR      0        0    10712       0       0        0     0    0    0   0   0   0
HAU      0    22899        0       0       0        0     0    0    0   0   0   0
HON      0     3156        0       0       0     5515     0    0    0   0   0   0
HOR      0        0        0   10424       0        0     0    0    0   0   0   0
JES      0        0        0       0       0    10176     0    0    0   0   0   0
KNA      0     5848        0       0       0        0     0    0    0   0   0   0
KON      0    11296        0       0       0        0     0    0    0   0   0   0
KOP      0     5878        0       0       0        0     0    0    0   0   0   0
KOR      0        0        0    9142       0        0     0    0    0   0   0   0
KOU      0     5244        0       0       0        0     0    0    0   0   0   0
KOV      0        0        0       0       0     6118     0    0    0   0   0   0
KRI      0        0        0   32567       0        0     0    0    0   0   0   0
```

Figure 4.1.3.2. Circle K stand-alone text file

```
TextCi.txt
KRU     0       0    9290       0        0        0    0    0    0    0    0    0
LAR     0       0       0   12531        0        0    0    0    0    0    0    0
LEI     0    7182       0       0        0        0    0    0    0    0    0    0
LEV     0       0    5253       0        0        0    0    0    0    0    0    0
LIL     0       0   10463       0        0        0    0    0    0    0    0    0
LOM     0       0    5694       0        0        0    0    0    0    0    0    0
MAN     0       0       0    5546        0        0    0    0    0    0    0    0
MOI     0       0    9608       0        0        0    0    0    0    0    0    0
MOL     0       0   10729       0        0        0    0    0    0    0    0    0
MOS     0       0       0   23964        0        0    0    0    0    0    0    0
NAR     0       0    7193       0        0        0    0    0    0    0    0    0
NES     0       0       0       0     6648        0    0    0    0    0    0    0
OSL     0       0   37471       0   313715   167140    0    0    0    0    0    0
OSO     0    7072       0       0        0        0    0    0    0    0    0    0
POR     0       0       0   47411        0        0    0    0    0    0    0    0
RAH     0       0       0       0        0     6866    0    0    0    0    0    0
SAN     0       0       0   22557        0        0    0    0    0    0    0    0
SKI     0       0       0       0     9937        0    0    0    0    0    0    0
STA     0   79365       0   35036        0        0    0    0    0    0    0    0
STE     0       0    6602       0        0        0    0    0    0    0    0    0
STJ     0       0    7485       0        0        0    0    0    0    0    0    0
TON     0       0       0   26650        0        0    0    0    0    0    0    0
TRD     0       0    8835       0        0        0    0    0    0    0    0    0
TRH     0       0       0       0        0    94748    0    0    0    0    0    0
TRO     0       0   20576       0        0        0    0    0    0    0    0    0
VEN     0       0       0    6669        0        0    0    0    0    0    0    0

:   SHKIR SHLAR SHLIL SHSJU SHSKJ SHTAN SHVES YXMOI YXSTA    :=
AAS   0     0     0     0     0     0     0     0     0
ALE   0     0     0     0     0     0     0     0     0
ALG   0     0     0     0     0     0     0     0     0
ALT   0     0     0     0     0     0     0     0     0
ARE   0     0     0     0     0     0     0     0     0
ASK   0     0     0     0     0     0     0     0     0
ASO   0     0     0     0     0     0     0     0     0
BER   0     0     0     0     0     0     0     0     0
BOD   0     0     0     0     0     0     0     0     0
BRU   0     0     0     0     0     0     0     0     0
BRY   0     0     0     0     0     0     0     0     0
DRA   0     0     0     0     0     0     0     0     0
DRO   0     0     0     0     0     0     0     0     0
EGE   0     0     0     0     0     0     0     0     0
```

Figure 4.1.3.3. Circle K stand-alone text file

```
TextCi.txt
ELV   0     0     0     0     0     0     0     0     0
FOR   0     0     0     0     0     0     0     0     0
FRE   0     0     0     0     0     0     0     0     0
GJO   0     0     0     0     0     0     0     0     0
GRI   0     0     0     0     0     0     0     0     0
HAL   0     0     0     0     0     0     0     0     0
HAM   0     0     0     0     0     0     0     0     0
HAR   0     0     0     0     0     0     0     0     0
HAU   0     0     0     0     0     0     0     0     0
HON   0     0     0     0     0     0     0     0     0
HOR   0     0     0     0     0     0     0     0     0
JES   0     0     0     0     0     0     0     0     0
KNA   0     0     0     0     0     0     0     0     0
KON   0     0     0     0     0     0     0     0     0
KOP   0     0     0     0     0     0     0     0     0
KOR   0     0     0     0     0     0     0     0     0
KOU   0     0     0     0     0     0     0     0     0
KOV   0     0     0     0     0     0     0     0     0
KRI   0     0     0     0     0     0     0     0     0
KRU   0     0     0     0     0     0     0     0     0
LAR   0     0     0     0     0     0     0     0     0
LEI   0     0     0     0     0     0     0     0     0
LEV   0     0     0     0     0     0     0     0     0
LIL   0     0     0     0     0     0     0     0     0
LOM   0     0     0     0     0     0     0     0     0
MAN   0     0     0     0     0     0     0     0     0
MOI   0     0     0     0     0     0     0     0     0
MOL   0     0     0     0     0     0     0     0     0
MOS   0     0     0     0     0     0     0     0     0
NAR   0     0     0     0     0     0     0     0     0
NES   0     0     0     0     0     0     0     0     0
OSL   0     0     0     0     0     0     0     0     0
OSO   0     0     0     0     0     0     0     0     0
POR   0     0     0     0     0     0     0     0     0
RAH   0     0     0     0     0     0     0     0     0
SAN   0     0     0     0     0     0     0     0     0
SKI   0     0     0     0     0     0     0     0     0
STA   0     0     0     0     0     0     0     0     0
STE   0     0     0     0     0     0     0     0     0
STJ   0     0     0     0     0     0     0     0     0
TON   0     0     0     0     0     0     0     0     0
TRD   0     0     0     0     0     0     0     0     0
```

Figure 4.1.3.4. Circle K stand-alone text file

```
TRD  0    0    0    0    0    0    0    0    0
TRH  0    0    0    0    0    0    0    0    0
TRO  0    0    0    0    0    0    0    0    0
VEN  0    0    0    0    0    0    0    0    0
;
```

Shell

Here we can see that only Shell is distributing fuel to their gas stations at their demands from their depots within their capacities.

Figure 4.1.3.5. Shell stand-alone text file

```
TextSh.txt
z = 126062000

f [*,*] (tr)
# $2 = CIFOR
# $3 = CIHAR
# $4 = CIKRI
# $5 = CIOSL
# $6 = CITRH
# $7 = EQMON
# $8 = ESBER
# $9 = ESFRE
# $10 = ESSLA
# $11 = ESTRH
# $13 = SHKIR
:   CIALT  $2  $3  $4  $5  $6  $7  $8  $9 $10 $11   SHBAL $13   SHLAR   SHLIL :=
AAS    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
ALE    0    0   0   0   0   0   0   0   0   0   0       0   0   27075       0
ALG    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
ALT    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
ARE    0    0   0   0   0   0   0   0   0   0   0       0   0       0   22123
ASK    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
ASO    0    0   0   0   0   0   0   0   0   0   0       0   0       0   11230
BER    0    0   0   0   0   0   0   0   0   0   0       0   0       0  130702
BOD    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
BRU    0    0   0   0   0   0   0   0   0   0   0       0   0    5420       0
BRY    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
DRA    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
DRO    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
EGE    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
ELV    0    0   0   0   0   0   0   0   0   0   0       0   0    7686       0
FOR    0    0   0   0   0   0   0   0   0   0   0       0   0    5256       0
FRE    0    0   0   0   0   0   0   0   0   0   0       0   0       0    4423
GJO    0    0   0   0   0   0   0   0   0   0   0       0   0   10468       0
GRI    0    0   0   0   0   0   0   0   0   0   0       0   0       0       0
HAL    0    0   0   0   0   0   0   0   0   0   0       0   0       0   13070
HAM    0    0   0   0   0   0   0   0   0   0   0       0   0   14208       0
HAR    0    0   0   0   0   0   0   0   0   0   0   10712   0       0       0
HAU    0    0   0   0   0   0   0   0   0   0   0       0   0       0   22899
HON    0    0   0   0   0   0   0   0   0   0   0       0   0    8671       0
HOR    0    0   0   0   0   0   0   0   0   0   0       0   0       0   10424
JES    0    0   0   0   0   0   0   0   0   0   0       0   0   10176       0
KNA    0    0   0   0   0   0   0   0   0   0   0       0   0       0    5848
```

Figure 4.1.3.6. Shell stand-alone text file

TextSh.txt

KON	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KOP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5878
KOR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KOU	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KOV	0	0	0	0	0	0	0	0	0	0	0	0	0	6118	0
KRI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
KRU	0	0	0	0	0	0	0	0	0	0	0	0	0	9290	0
LAR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12531
LEI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7182
LEV	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
LIL	0	0	0	0	0	0	0	0	0	0	0	0	0	10463	0
LOM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MAN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MOI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MOL	0	0	0	0	0	0	0	0	0	0	0	0	0	10729	0
MOS	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
NAR	0	0	0	0	0	0	0	0	0	0	0	7193	0	0	0
NES	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
OSL	0	0	0	0	0	0	0	0	0	0	0	0	0	201893	0
OSO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7072
POR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	47411
RAH	0	0	0	0	0	0	0	0	0	0	0	0	0	6866	0
SAN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22557
SKI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
STA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
STE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
STJ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TON	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26650
TRD	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
TRH	0	0	0	0	0	0	0	0	0	0	0	0	0	15681	0
TRO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
VEN	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

:	SHSJU	SHSKJ	SHTAN	SHVES	YXMOI	YXSTA	:=
AAS	0	0	5525	0	0	0	
ALE	0	0	0	0	0	0	
ALG	0	0	5763	0	0	0	
ALT	0	7800	0	0	0	0	
ARE	0	0	0	0	0	0	
ASK	7366	0	0	0	0	0	
ASO	0	0	0	0	0	0	
BER	0	0	0	0	0	0	

Figure 4.1.3.7. Shell stand-alone text file

TextSh.txt

BOD	0	0	0	21211	0	0
BRU	0	0	0	0	0	0
BRY	0	0	6204	0	0	0
DRA	0	0	54872	0	0	0
DRO	0	0	6809	0	0	0
EGE	0	0	5813	0	0	0
ELV	0	0	0	0	0	0
FOR	0	0	0	0	0	0
FRE	16264	0	37079	0	0	0
GJO	0	0	0	0	0	0
GRI	0	0	6764	0	0	0
HAL	0	0	0	0	0	0
HAM	0	0	0	0	0	0
HAR	0	0	0	0	0	0
HAU	0	0	0	0	0	0
HON	0	0	0	0	0	0
HOR	0	0	0	0	0	0
JES	0	0	0	0	0	0
KNA	0	0	0	0	0	0
KON	0	0	11296	0	0	0
KOP	0	0	0	0	0	0
KOR	0	0	9142	0	0	0
KOU	0	0	5244	0	0	0
KOV	0	0	0	0	0	0
KRI	0	0	32567	0	0	0
KRU	0	0	0	0	0	0
LAR	0	0	0	0	0	0
LEI	0	0	0	0	0	0
LEV	0	0	0	5253	0	0
LIL	0	0	0	0	0	0
LOM	0	0	5694	0	0	0
MAN	0	0	5546	0	0	0
MOI	0	0	0	9608	0	0
MOL	0	0	0	0	0	0
MOS	0	0	23964	0	0	0
NAR	0	0	0	0	0	0
NES	0	0	6648	0	0	0
OSL	316433	0	0	0	0	0
OSO	0	0	0	0	0	0
POR	0	0	0	0	0	0
RAH	0	0	0	0	0	0
SAN	0	0	0	0	0	0

Figure 4.1.3.8. Shell stand-alone text file

```
SKI    9937       0       0       0    0     0
STA       0       0  114401       0    0     0
STE       0       0       0    6602    0     0
STJ       0       0       0    7485    0     0
TON       0       0       0       0    0     0
TRD       0    8835       0       0    0     0
TRH       0       0       0   79067    0     0
TRO       0   20576       0       0    0     0
VEN       0       0    6669       0    0     0
;
```

Esso

Here we can see that only Esso is distributing fuel to their gas stations at their demands from their depots within their capacities.

Figure 4.1.3.9. Esso stand-alone text file

TextEs.txt

```
z = 48454000

f [*,*] (tr)
# $2 = CIFOR
# $3 = CIHAR
# $4 = CIKRI
# $5 = CIOSL
# $6 = CITRH
# $7 = EQMON
# $12 = SHBAL
# $13 = SHKIR
# $14 = SHLAR
:   CIALT  $2  $3  $4  $5  $6  $7   ESBER   ESFRE    ESSLA    ESTRH $12 $13 $14 :=
AAS    0    0   0   0   0   0   0       0       0     3825        0   0   0   0
ALE    0    0   0   0   0   0   0       0       0        0    18744   0   0   0
ALG    0    0   0   0   0   0   0    3990       0        0        0   0   0   0
ALT    0    0   0   0   0   0   0       0       0        0     5400   0   0   0
ARE    0    0   0   0   0   0   0       0       0    15316        0   0   0   0
ASK    0    0   0   0   0   0   0       0    5099        0        0   0   0   0
ASO    0    0   0   0   0   0   0    7774       0        0        0   0   0   0
BER    0    0   0   0   0   0   0   90486       0        0        0   0   0   0
BOD    0    0   0   0   0   0   0       0       0        0    14684   0   0   0
BRU    0    0   0   0   0   0   0       0    3752        0        0   0   0   0
BRY    0    0   0   0   0   0   0    4295       0        0        0   0   0   0
DRA    0    0   0   0   0   0   0       0       0    37988        0   0   0   0
DRO    0    0   0   0   0   0   0       0    4714        0        0   0   0   0
EGE    0    0   0   0   0   0   0    4024       0        0        0   0   0   0
ELV    0    0   0   0   0   0   0       0    5321        0        0   0   0   0
FOR    0    0   0   0   0   0   0    3639       0        0        0   0   0   0
FRE    0    0   0   0   0   0   0       0   39992        0        0   0   0   0
GJO    0    0   0   0   0   0   0       0       0     7247        0   0   0   0
GRI    0    0   0   0   0   0   0       0       0     4683        0   0   0   0
HAL    0    0   0   0   0   0   0       0    9049        0        0   0   0   0
HAM    0    0   0   0   0   0   0       0    9837        0        0   0   0   0
HAR    0    0   0   0   0   0   0       0       0        0     7416   0   0   0
HAU    0    0   0   0   0   0   0   15853       0        0        0   0   0   0
HON    0    0   0   0   0   0   0       0       0     6003        0   0   0   0
HOR    0    0   0   0   0   0   0       0       0     7217        0   0   0   0
JES    0    0   0   0   0   0   0       0    7045        0        0   0   0   0
KNA    0    0   0   0   0   0   0    4048       0        0        0   0   0   0
KON    0    0   0   0   0   0   0       0       0     7820        0   0   0   0
KOP    0    0   0   0   0   0   0    4069       0        0        0   0   0   0
```

Figure 4.1.3.10. Esso stand-alone text file

TextEs.txt

```
KOR  0  0  0  0  0  0  0      0       0    6329     0  0  0  0
KOU  0  0  0  0  0  0  0      0       0    3630     0  0  0  0
KOV  0  0  0  0  0  0  0      0    4236       0     0  0  0  0
KRI  0  0  0  0  0  0  0      0       0   22546     0  0  0  0
KRU  0  0  0  0  0  0  0      0       0       0  6432  0  0  0
LAR  0  0  0  0  0  0  0      0       0    8675     0  0  0  0
LEI  0  0  0  0  0  0  0   4972       0       0     0  0  0  0
LEV  0  0  0  0  0  0  0      0       0       0  3637  0  0  0
LIL  0  0  0  0  0  0  0      0    7244       0     0  0  0  0
LOM  0  0  0  0  0  0  0      0       0    3942     0  0  0  0
MAN  0  0  0  0  0  0  0      0       0    3840     0  0  0  0
MOI  0  0  0  0  0  0  0      0       0       0  6652  0  0  0
MOL  0  0  0  0  0  0  0      0       0       0  7428  0  0  0
MOS  0  0  0  0  0  0  0      0       0   16590     0  0  0  0
NAR  0  0  0  0  0  0  0      0       0       0  4980  0  0  0
NES  0  0  0  0  0  0  0      0    4602       0     0  0  0  0
OSL  0  0  0  0  0  0  0      0  244356  114485     0  0  0  0
OSO  0  0  0  0  0  0  0   4896       0       0     0  0  0  0
POR  0  0  0  0  0  0  0      0       0   32823     0  0  0  0
RAH  0  0  0  0  0  0  0      0    4753       0     0  0  0  0
SAN  0  0  0  0  0  0  0      0       0   15616     0  0  0  0
SKI  0  0  0  0  0  0  0      0       0    6880     0  0  0  0
STA  0  0  0  0  0  0  0  79201       0       0     0  0  0  0
STE  0  0  0  0  0  0  0      0       0       0  4570  0  0  0
STJ  0  0  0  0  0  0  0      0       0       0  5182  0  0  0
TON  0  0  0  0  0  0  0      0       0   18450     0  0  0  0
TRD  0  0  0  0  0  0  0      0       0       0  6116  0  0  0
TRH  0  0  0  0  0  0  0      0       0       0 65595  0  0  0
TRO  0  0  0  0  0  0  0      0       0       0 14245  0  0  0
VEN  0  0  0  0  0  0  0      0       0    4617     0  0  0  0

:    SHLIL SHSJU SHSKJ SHTAN SHVES YXMOI YXSTA    :=
AAS    0     0     0     0     0     0     0
ALE    0     0     0     0     0     0     0
ALG    0     0     0     0     0     0     0
ALT    0     0     0     0     0     0     0
ARE    0     0     0     0     0     0     0
ASK    0     0     0     0     0     0     0
ASO    0     0     0     0     0     0     0
BER    0     0     0     0     0     0     0
BOD    0     0     0     0     0     0     0
BRU    0     0     0     0     0     0     0
```

Figure 4.1.3.11. Esso stand-alone text file

TextEs.txt

```
BRY    0     0     0     0     0     0     0
DRA    0     0     0     0     0     0     0
DRO    0     0     0     0     0     0     0
EGE    0     0     0     0     0     0     0
ELV    0     0     0     0     0     0     0
FOR    0     0     0     0     0     0     0
FRE    0     0     0     0     0     0     0
GJO    0     0     0     0     0     0     0
GRI    0     0     0     0     0     0     0
HAL    0     0     0     0     0     0     0
HAM    0     0     0     0     0     0     0
HAR    0     0     0     0     0     0     0
HAU    0     0     0     0     0     0     0
HON    0     0     0     0     0     0     0
HOR    0     0     0     0     0     0     0
JES    0     0     0     0     0     0     0
KNA    0     0     0     0     0     0     0
KON    0     0     0     0     0     0     0
KOP    0     0     0     0     0     0     0
KOR    0     0     0     0     0     0     0
KOU    0     0     0     0     0     0     0
KOV    0     0     0     0     0     0     0
KRI    0     0     0     0     0     0     0
KRU    0     0     0     0     0     0     0
LAR    0     0     0     0     0     0     0
LEI    0     0     0     0     0     0     0
LEV    0     0     0     0     0     0     0
LIL    0     0     0     0     0     0     0
LOM    0     0     0     0     0     0     0
MAN    0     0     0     0     0     0     0
MOI    0     0     0     0     0     0     0
MOL    0     0     0     0     0     0     0
MOS    0     0     0     0     0     0     0
NAR    0     0     0     0     0     0     0
NES    0     0     0     0     0     0     0
OSL    0     0     0     0     0     0     0
OSO    0     0     0     0     0     0     0
POR    0     0     0     0     0     0     0
RAH    0     0     0     0     0     0     0
SAN    0     0     0     0     0     0     0
SKI    0     0     0     0     0     0     0
STA    0     0     0     0     0     0     0
```

Figure 4.1.3.12. Esso stand-alone text file

```
STA   0   0   0   0   0   0   0
STE   0   0   0   0   0   0   0
STJ   0   0   0   0   0   0   0
TON   0   0   0   0   0   0   0
TRD   0   0   0   0   0   0   0
TRH   0   0   0   0   0   0   0
TRO   0   0   0   0   0   0   0
VEN   0   0   0   0   0   0   0
;
```

YX

Here we can see that only YX is distributing fuel to their gas stations at their demands from their depots within their capacities.

Figure 4.1.3.13. YX stand-alone text file

```
TextYX.txt
z = 97234400

f [*,*] (tr)
# $1 = CIALT
# $2 = CIFOR
# $3 = CIHAR
# $4 = CIKRI
# $5 = CIOSL
# $6 = CITRH
# $7 = EQMON
# $8 = ESBER
# $9 = ESFRE
# $10 = ESSLA
# $11 = ESTRH
# $12 = SHBAL
# $13 = SHKIR
# $14 = SHLAR
# $15 = SHLIL
# $16 = SHSJU
# $17 = SHSKJ
# $18 = SHTAN
# $19 = SHVES
:    $1  $2  $3  $4  $5  $6  $7  $8  $9 $10 $11 $12 $13 $14 $15 $16 $17 $18 $19 :=
AAS   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ALE   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ALG   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ALT   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ARE   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ASK   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ASO   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
BER   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
BOD   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
BRU   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
BRY   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
DRA   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
DRO   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
EGE   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
ELV   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
FOR   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
FRE   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
GJO   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
GRI   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
```

Figure 4.1.3.14. YX stand-alone text file

```
TextYX.txt
HAL   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
HAM   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
HAR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
HAU   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
HON   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
HOR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
JES   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KNA   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KON   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KOP   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KOR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KOU   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KOV   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KRI   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
KRU   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
LAR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
LEI   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
LEV   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
LIL   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
LOM   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
MAN   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
MOI   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
MOL   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
MOS   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
NAR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
NES   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
OSL   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
OSO   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
POR   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
RAH   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
SAN   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
SKI   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
STA   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
STE   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
STJ   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
TON   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
TRD   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
TRH   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
TRO   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
VEN   0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
```

Figure 4.1.3.15. YX stand-alone text file

```
TextYX.txt
:      YXMOI   YXSTA    :=
AAS        0    2125
ALE    10413       0
ALG        0    2216
ALT     3000       0
ARE        0    8509
ASK     2833       0
ASO        0    4319
BER        0   50270
BOD     8158       0
BRU     2084       0
BRY        0    2386
DRA        0   21105
DRO        0    2619
EGE        0    2236
ELV     2956       0
FOR        0    2022
FRE        0   22218
GJO     4026       0
GRI        0    2601
HAL        0    5027
HAM     5465       0
HAR     4120       0
HAU        0    8807
HON        0    3335
HOR        0    4009
JES     3914       0
KNA        0    2249
KON        0    4345
KOP        0    2261
KOR        0    3516
KOU        0    2017
KOV     2353       0
KRI        0   12526
KRU     3573       0
LAR        0    4819
LEI        0    2762
LEV     2021       0
LIL     4024       0
LOM        0    2190
MAN        0    2133
```

Figure 4.1.3.16. YX stand-alone text file

```
MOI     3696        0
MOL     4126        0
MOS        0     9217
NAR     2767        0
NES        0     2557
OSL   130199    69157
OSO        0     2720
POR        0    18235
RAH     2641        0
SAN        0     8676
SKI     3822        0
STA        0    44001
STE     2539        0
STJ     2879        0
TON        0    10250
TRD     3398        0
TRH    36442        0
TRO     7914        0
VEN        0     2565
;
```

Cost results

Circle K

Circle K's stand-alone cost

Figure 4.1.3.17. Circle K stand-alone text file

```
TextCi.txt
z = 131161000
```

Shell

Shell's stand-alone cost

Figure 4.1.3.18. Shell stand-alone text file

```
TextSh.txt
z = 126062000
```

Esso

Esso's stand-alone cost

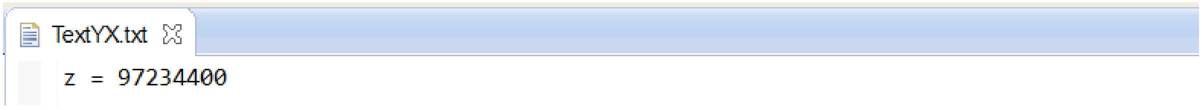
Figure 4.1.3.19. Esso stand-alone text file

```
TextEs.txt
z = 48454000
```

YX

YX's stand-alone cost

Figure 4.1.3.20. YX stand-alone text file



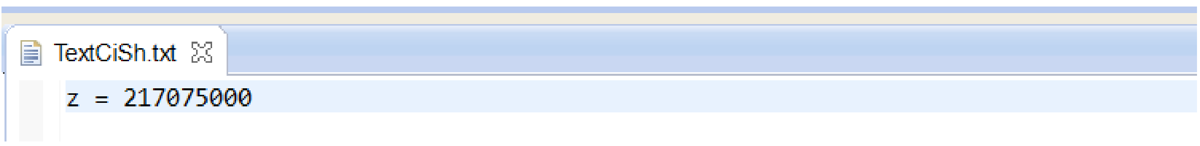
4.2 Two companies collaboration analysis

In two companies collaboration I investigate the costs for two and two companies collaborates in a best possible way regarding achieve lowest possible total cost for two companies distributing their fuel from these two companies depots to their gas stations with their corresponding demands. In the previous chapters I have shown what sort of changes I have done in the data files and the run files at the different analyses. I will in the following therefor only show the results regarding cost from the text files from the different analyses I have done, as the changes will be equivalently here as already described above.

Cost results

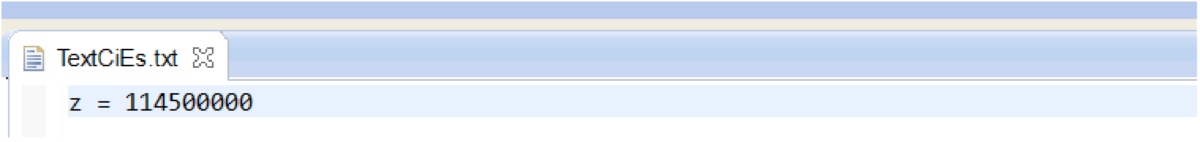
Circle K / Shell

Figure 4.2.1. Circle K / Shell coalition text file



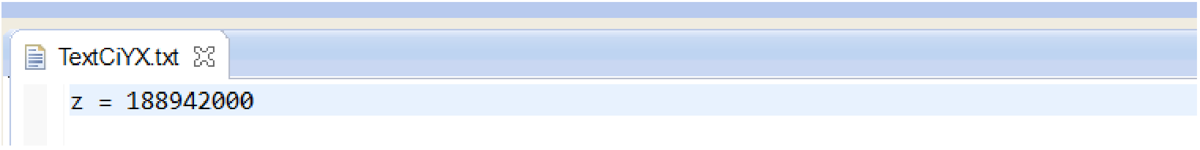
Circle K / Esso

Figure 4.2.2. Circle K / Esso coalition text file



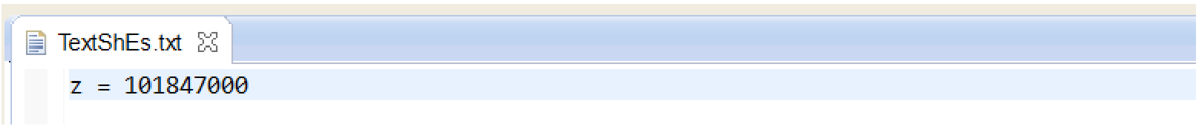
Circle K / YX

Figure 4.2.3. Circle K / YX coalition text file



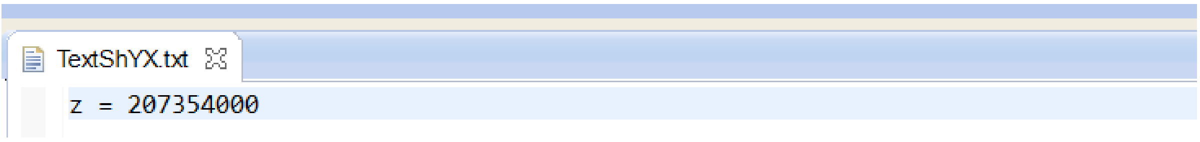
Shell / Esso

Figure 4.2.4. Shell / Esso coalition text file



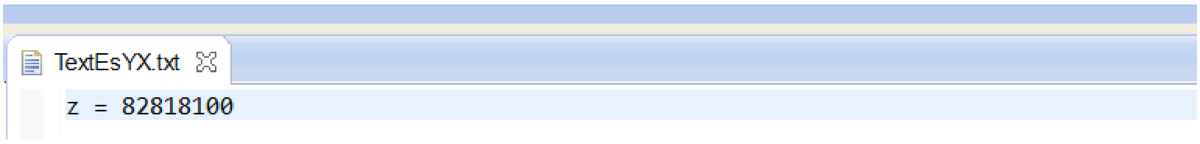
Shell / YX

Figure 4.2.5. Shell / YX coalition text file



Esso / YX

Figure 4.2.6. Esso / YX coalition text file

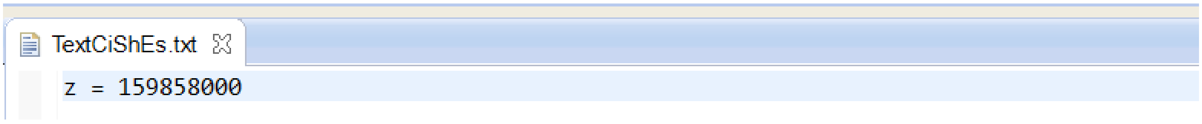


4.3 Three companies collaboration analysis

Cost results

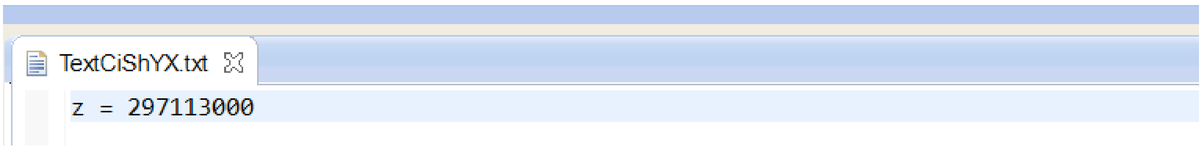
Circle K / Shell / Esso

Figure 4.3.1. Circle K / Shell / Esso coalition text file



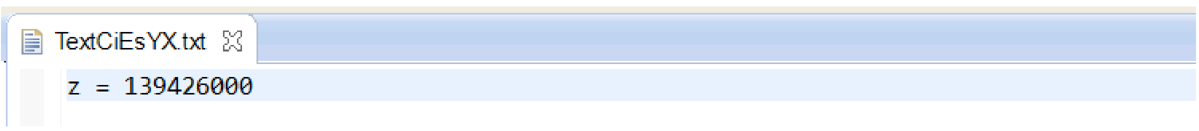
Circle K / Shell / YX

Figure 4.3.2. Circle K / Shell / YX coalition text file



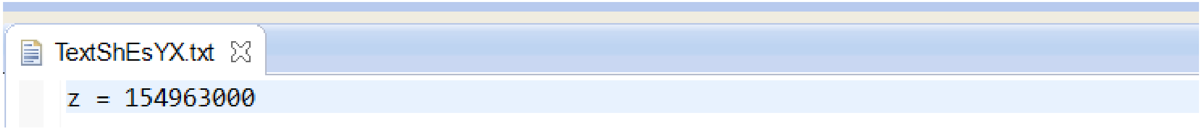
Circle K / Esso / YX

Figure 4.3.3. Circle K / Esso / YX coalition text file



Shell / Esso / YX

Figure 4.3.4. Shell / Esso / YX coalition text file

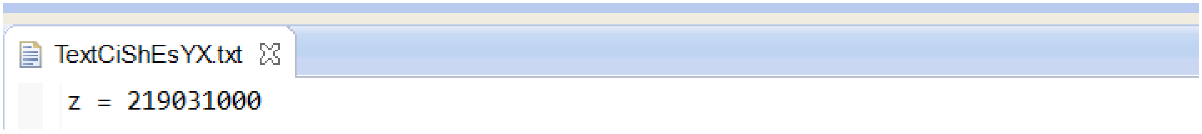


4.4 Four companies(grand coalition) collaboration analysis

Cost results

Circle K / Shell / Esso / YX

Figure 4.4.1. Circle K / Shell / Esso / YX coalition text file



Circle K / Shell / Esso / YX / Equinor depot

Figure 4.4.2. Circle K / Shell / Esso / YX coalition plus Equinor depot text file

5. AMPL analyses and calculations with results

I will in this chapter run three series of 16 analyses in AMPL, which is 3 different approaches regarding depot capacities, to investigate if there is a pattern in what the results will show. The differences in the three series will be in the depots capacities, due to uncertain data for this parameter. In this way I hope to find a pattern which is similar in all the three approaches and then be able to make more reliable conclusions from the results. A serie of 15 analyses will give costs for each possible coalition including stand alone. These 15 analyses is all needed for allocating the grand coalition's cost to each company in Shapley Values. The 16th analysis will give the cost for the grand coalition were Equinor's depot is included.

I will also do the calculations regarding the grand coalition's cost allocation in Shapley Values.

Further I will check the three condition tests for each of the three series to find out if the grand coalition is the most beneficial for all the four companies.

In addition I will run 5 analyses in AMPL regarding CO^2 emission for one serie and calculate the CO^2 emission for all three series.

This will in total include 53 analyses in AMPL plus corresponding calculations.

From these results I will calculate the savings and decreased CO^2 emission in each of the three series from optimized collaboration.

5.1 Fuel distribution with equal depot capacities

Here I have given all depots an equal capacity which is the total annual consumption of fuel in Norway divided by the total number of depots multiplied with approximately 1.5 to give overcapacities at a middle extent.

Total annual fuel consumption in Norway in 1000 litre: 4874893

Total number of fuel depots in Norway including Mongstad: 21

4874893 / 21 x 1.5 = 348206.6

Depot capacities in 1000 litre for each depot: 350000

5.1.1 Coalition transportation cost results from AMPL listed

The costs showed below is the transportation costs from AMPL for all possible coalitions in this game for the four companies. This do not mean the cost for each company (except for stand alone), but the cost for the entire coalition.

Circle K	131161000
Shell	126062000
Esso	48454000
YX	97234400
Circle K / Shell	217075000
Circle K / Esso	114500000
Circle K / YX	188942000
Shell / Esso	101847000
Shell / YX	207354000
Esso / YX	82818100
Circle K / Shell / Esso	159858000
Circle K / Shell / YX	297113000
Circle K / Esso / YX	139426000
Shell / Esso / YX	154963000
Circle K / Shell / Esso / YX	219031000
Circle K / Shell / Esso / YX / Equinor depot	206766000

5.1.2 Savings for each coalition

Here I will find the savings in percentage for each coalition. To find this I compare the cost for the coalition to the companies summed stand-alone costs of the corresponding coalition.

Circle K / Shell

Summed stand-alone costs = 131161000 + 126062000 = 257223000

Coalition cost = 217075000

Savings = (257223000 – 217075000) / 257223000 = 15.6 %

Circle K / Esso

Summed stand-alone costs = 131161000 + 82818100 = 213979100

Coalition cost = 114500000

Savings = (213979100 – 114500000) / 213979100 = 46.5 %

Circle K / YX

Summed stand-alone costs = 131161000 + 97234400 = 228395400

Coalition cost = 188942000

Savings = (228395400 – 188942000) / 228395400 = 17.3 %

Shell / Esso

Summed stand-alone costs = 126062000 + 48454000 = 174516000

Coalition cost = 101847000

Savings = (174516000 – 101847000) / 174516000 = 41.6 %

Shell / YX

Summed stand-alone costs = 126062000 + 97234400 = 223296400

Coalition cost = 207354000

Savings = (223296400 – 207354000) / 223296400 = 7.1 %

Esso / YX

Summed stand-alone costs = 48454000 + 97234400 = 145688400

Coalition cost = 82818100

Savings = (145688400 – 82818100) / 145688400 = 43.2 %

Circle K / Shell / Esso

Summed stand-alone costs = 131161000 + 126062000 + 48454000 = 305677000

Coalition cost = 159858000

Savings = (305677000 – 159858000) / 305677000 = 47.7 %

Circle K / Shell / YX

Summed stand-alone costs = 131161000 + 126062000 + 97234400 = 354457400

Coalition cost = 297113000

Savings = (354457400 – 297113000) / 354457400 = 16.2 %

Circle K / Esso / YX

Summed stand-alone costs = 131161000 + 48454000 + 97234400 = 276849400

Coalition cost = 139426000

Savings = (276849400 – 139426000) / 276849400 = 49.6 %

Shell / Esso / YX

Summed stand-alone costs = 126062000 + 48454000 + 97234400 = 271750400

Coalition cost = 154963000

Savings = (271750400 – 154963000) / 271750400 = 43.0 %

Circle K / Shell / Esso / YX

Summed stand-alone costs = 131161000 + 126062000 + 48454000 + 97234400 = 402911400

Coalition cost = 219031000

Savings = 402911400 NOK – 219031000 NOK = 183880400 NOK

Savings = (402911400 – 219031000) / 402911400 = 45.6 %

Circle K / Shell / Esso / YX / Equinor depot

This case must be seen as a special case as Equinor depot do not have a stand-alone cost, due to they do not own or operate any gas stations. However, I want to include this analysis to get a picture of what savings can be done in NOK and CO^2 emission when the companies can choose also to distribute fuel from this depot to their gas stations.

Summed stand-alone costs = 131161000 + 126062000 + 48454000 + 97234400 = 402911400

Coalition cost = 206766000

Savings = 402911400 NOK – 206766000 NOK = 196145400 NOK

Savings = (402911400 – 206766000) / 402911400 = 48.7 %

5.1.3 Game theory

First I will use the cost allocation tool Shapley Values to allocate the coalition cost to each player (company) in this 4-player game. This formula belongs to cooperative game theory, which I described in the Theory chapter, regarding calculation and the mechanism behind the formula.

The reason I choose Shapley Values method for this cost allocation is that it weight each players contribution to the collaboration in a good and accurate way, which is likely to give a fair cost allocation. Shapley Values is also in my opinion a well-known and trusted method for cost allocation.

After I have calculated the cost allocation from Shapley Values I will further check the three conditions, to find out if each player is most beneficial to join the full collaboration (grand coalition) or if any of the players are more beneficial to form smaller coalitions or stand alone, given the cost allocation. This is calculated in the chapters below the chapter "Shapley Values".

This cost allocation belongs to the core if there are no other(smaller) coalitions that can be more beneficial than this 4 player coalition (grand coalition) in this game. This cost allocation is stable if none of the 4 players will have benefit from braking out to form smaller coalitions or stand alone.

To find out if the cost allocation is stable and belongs to the core I will check the three conditions individual rational, coalition rational and efficiency condition.

5.1.4 Shapley Values

To set up the calculations in the Shapley Values formula I here uses all the 15 coalition costs (included the stand alone costs) from the analyses in AMPL with equal depot capacities.

Regarding the grand coalition cost I will here use the result from the analysis where Equinor depot is not included, as Equinor was not included in the other analyses. If I here had used the grand coalition result where Equinor's depot is included, the Shapley values calculation had been fault, as Equinor's depot is not included in the smaller coalition analyses and the stand-alone analyses. Therefore I need to be consistence when calculating Shapley values to get correct values from the formula.

These cost results are calculated according to the cost allocation method Shapley Value, which I already have described in detail in the Theory chapter at page 18-22, both for calculation and logic behind Shapley Values. I will therefore not show calculations for Shapley Values here.

$$\varphi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\;(n-|S|-1)!}{n!}(v(S \cup \{i\}) - v(S))$$

The results(allocated costs) from the Shapley Values formula in the game with equal depot capacities are listed below.

Circle K: 86569291.67

Shell: 91008458.33

Esso: -25275025.00

YX: 66728275.00

5.1.5 Individual rational

Here I will check if these Shapley values are individual rational, by checking if the calculated Shapley value for each player (company) is lower than their stand-alone value from AMPL. This is the first step to find out if the cost allocation is stable and belongs to the core.

Circle K: 86569291.67 < 131161000 (yes)

Shell: 91008458.33 < 126062000 (yes)

Esso: -25275025.00 < 48454000 (yes)

YX: 66728275.00 < 97234400 (yes)

5.1.6 Coalition rational

Here I will check if the calculated Shapley values for every coalition summed are lower than their respective coalition cost from AMPL. The coalition for all 4 companies will of course not be calculated here, only the 2 and 3 player coalitions. The 4 companies coalition will be tested in another test in the next section.

Circle K / Shell: (86569291.67 + 91008458,33) < 217075000 (yes)

Circle K / Esso: (86569291.67 + (-25275025,00)) < 114500000 (yes)

Circle K / YX: (86569291.67 + 66728275,00) < 188942000 (yes)

Shell / Esso: (91008458.33 + (-25275025,00)) < 101847000 (yes)

Shell / YX: (91008458.33 + 66728275,00 < 207354000 (yes)

Esso / YX: (-25275025.00 + 66728275,00) < 82818100 (yes)

Circle K / Shell / Esso: (86569291.67 + 91008458,33 + (-25275025.00)) < 159858000 (yes)

Circle K / Shell / YX: (86569291.67 + 91008458,33 + 66728275.00) < 297113000 (yes)

Circle K / Esso / YX: (86569291.67 + (-25275025,00) + 66728275.00) < 139426000 (yes)

Shell / Esso / YX: (91008458.33 + (-25275025,00) + 66728275.00) < 154963000 (yes)

Since I only have yeses, the rational condition for coalitions are fulfilled.

5.1.7 Efficiency condition

A final test is the efficiency condition test, where I going to check if all 4 Shapley values summed are equal to the full coalition cost from AMPL.

Circle K / Shell / Esso / YX: (86569291.67 + 91008458.33 + (-25275025.00) + 66728275.00) = 219031000 (yes)

5.1.8 Conclusion from the individual rational-, coalition rational- and efficiency condition tests

This cost allocation from the Shapley values formula passed all three tests, which means according to game theory, that this cost allocation is stable and belongs to the core. Then we also know that non of the 4 companies have an economical benefit to break out of the 4-player coalition to form a smaller coalition. They all will benefit most when they stay in this grand coalition.

5.1.9 Savings for each company

Here I will calculate each company’s savings when comparing their Shapley values to their stand-alone costs. This will tell how much in annually NOK and in percentage each of the four companies saves to join this collaboration where all four companies collaborate together. As these calculations use the cost allocation from Shapley Values this will show results for the grand coalition when Equinor’s depot is not included.

Circle K: 131161000.00 – 86569291.67 = 44591708.33

Annually savings = 44591708.33 NOK

44591708.33 / 131161000,00 = 0.34

Percentage savings = 34.0 %

Shell: 126062000 – 91008458.33 = 35053541.67

Annually savings = 35053541.67 NOK

35053541.67 / 126062000 = 0.278

Percentage savings = 27.8 %

Esso: 48454000 – (-25275025.00) = 73729025

Annually savings = 73729025 NOK

73729025 / 48454000 = 1.522

Percentage savings = 152.2 %

YX: 97234400 – 66728275.00 = 30506125

Annually savings = 30506125 NOK

30506125 / 97234400 = 0.314

Percentage savings = 31.4 %

5.1.10 Reduction in CO^2 emission at full collaboration

Here I will find the annual reduction in CO^2 emissions from the trucks when the 4 companies collaborates together in the grand coalition compared to if they distribute fuel as stand-alone companies. I will first do some manual calculations.

From own knowledge and statements from people in the driver industry I suggest that a gas trucks tank contains when full tank typical 40000 litre and that a truck driving with full tank of that amount consumes at average approximately 0,4 litre diesel per km driving.

40000 / 1000 = 40

0.4 / 40 = 0.01

This gives that the trucks consumption of diesel is estimated to 0.01 litre diesel per 1000 litre km, given that they drive with full tanks of 40000 litre on the trucks.

The CO^2 emission from a diesel engine is 2660 gram per consumed litre of diesel (Helleborg, 2018).

2660 gram x 0,01 = 26,6 gram

This gives us that the trucks emission of CO^2 is 26.6 gram per 1000 litre km of distributed fuel.

26.6 gram = 0.0000266 ton

To find the annual reduction in CO^2 emission in ton there is a short cut that can be done. It is simply to divide 0.0000266 with the cost parameter c 0.31 in AMPL and then multiply this by the savings from the full collaboration, which here will be the version that includes Equinor's depot at Mongstad. The reason why I do include Equinor here is that this will not give any disturbance in the calculation as it would in Shapley values earlier. In this way I will get the results where the 4 companies also uses Equinor's depot, as I expect them to do, when seeking for the best possible way to collaborate and distribute fuel in Norway. Let's try to calculate.

0.0000266 / 0.31 = 0.00008581

0.00008581 x 196145400 = 16830.54

Annual reduction in CO^2 emission from full collaboration = 16830.54 ton

However, I still going to do this in some new analyses in AMPL to also make a functional program for CO^2 emission for this case.

I will now run 5 new analyses in AMPL to find the potential reduction in CO^2 at full collaboration. First I will run 4 new stand-alone analyses. Then I will run a last analysis with full collaboration including Equinor's depot.

To manage to run these new analyses I will now make 5 new data files, 5 new run files and 5 new text files in AMPL. I will also in this case use the same model file I already have created, which will be unchanged.

The only change I need to do in the data files compared to the previous relevant files are that I change the parameter c cost value 0.31 to 0.0000266 for CO^2 emission. The output will now show total annual CO^2 output from the corresponding company or companies in the analysis.

5.1.11 Creating CO^2 emission program in AMPL

Changes for CO^2 emission stand alone data files

Figure 5.1.11.1. Changes for the CO^2 emission stand-alone data files compared to the stand-alone data files for costs.

```
param c :=
# CO2 emission for transport of fuel per 1000L km in ton
0.0000266
;
```

Figure 5.1.11.2. CO^2 emission Circle K stand alone run file

DataCOCi.dat | RunCOCi.run | TextCOCi.txt

```
reset;
model Model.mod;
data DataCOCi.dat;
option solver cplex;
solve;
display z, f > TextCOCi.txt;
```

Figure 5.1.11.3. CO^2 emission Circle K stand alone text file

DataCOCi.dat | RunCOCi.run | TextCOCi.txt

```
z = 11254.5
```

The other CO^2 emission stand alone run files will have equivalently changes, I will therefore not show them here. Text files will give each new analysis individual results regarding CO^2.

Changes for CO^2 emission Circle K / Shell / Esso / YX / Equinor depot data file

Figure 5.1.11.4. Changes for CO^2 emission Circle K / Shell / Esso / YX coalition plus Equinor depot data file compared to Circle K / Shell / Esso / YX coalition plus Equinor depot data file for cost.

```
param c :=
# CO2 emission for transport of fuel per 1000L km in ton
0.0000266
;
```

Figure 5.1.11.5. CO^2 emission Circle K / Shell / Esso / YX coalition plus Equinor depot run file

```
DataCO.dat | RunCO.run | TextCO.txt

reset;
model Model.mod;
data DataCO.dat;
option solver cplex;
solve;
display z, f > TextCO.txt;
```

Figure 5.1.11.6. CO^2 emission Circle K / Shell / Esso / YX coalition plus Equinor depot text file

```
DataCO.dat | RunCO.run | TextCO.txt

z = 17741.8
```

CO^2 emission results

The results shows annual CO^2 emission from distributing fuel.

Figure 5.1.11.7. CO^2 emission Circle K stand alone text file

```
DataCOCi.dat | RunCOCi.run | TextCOCi.txt

z = 11254.5
```

Figure 5.1.11.8. CO^2 emission Shell stand alone text file

```
DataCOSh.dat | RunCOSh.run | TextCOSh.txt

z = 10816.9
```

Figure 5.1.11.9. CO^2 emission Esso stand alone text file

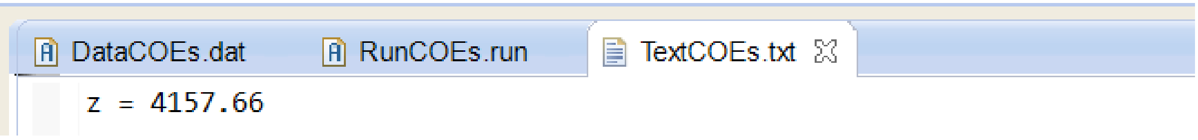


Figure 5.1.11.10. CO^2 emission YX stand alone text file

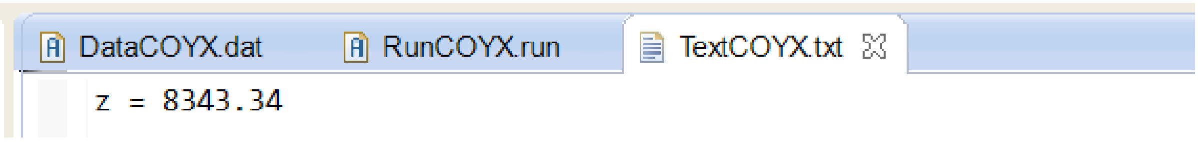
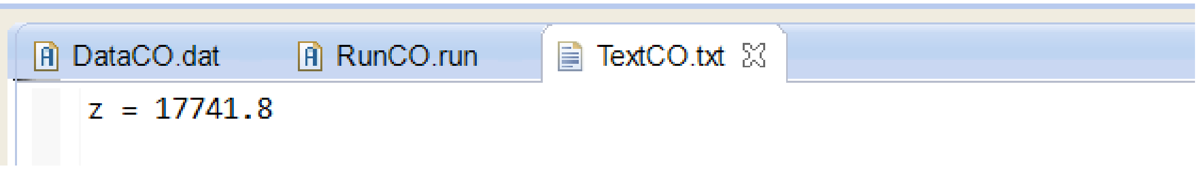


Figure 5.1.11.11. CO^2 emission Circle K / Shell / Esso / YX coalition plus Equinor depot text file



Circle K:	11254.5 ton CO^2
Shell:	10816.9 ton CO^2
Esso:	4157.66 ton CO^2
YX:	8343.34 ton CO^2
Sum:	34572.4 ton CO^2

Circle K / Shell / Esso / YX / Equinor depot: 17741.8 ton CO^2

34572.4 – 17741.8 = 16830.6

This means that a optimized collaborating in fuel distributing gives an annual reduction in CO^2 emission of 16830.6 ton CO^2 compared to if there is no collaboration. This is the same answer as in the short cut method above.

5.2 Fuel distribution with limitless depot capacities

Here I have set all depot capacities to 5000000 which is more than the total annual fuel consumption in Norway. This can then be seen as limitless depot capacities as every depot could in theory supply whole Norway with fuel on its own.

Total annual fuel consumption in Norway in 1000 litre: 4874893

Depots capacities: 5000000

The changes in AMPL files will be in the data files regarding depot capacities. Other than that the procedure for coding the AMPL program is all the same as I described for the serie with "equal depot capacities". Therefore I will here display the results I got from AMPL directly without screenshots.

5.2.1 Coalition transportation cost results from AMPL listed

Circle K	43511300
Shell	42164400
Esso	48063600
YX	73247100
Circle K / Shell	48416500
Circle K / Esso	48798900
Circle K / YX	44937700
Shell / Esso	31220400

Shell / YX	57276400
Esso / YX	51858900
Circle K / Shell / Esso	31895200
Circle K / Shell / YX	56136300
Circle K / Esso / YX	38757700
Shell / Esso / YX	36930300
Circle K / Shell / Esso / YX	34398700
Circle K / Shell / Esso / YX / Equinor depot	34398700

5.2.2 Savings for each coalition

Circle K / Shell

Summed stand-alone costs = 43511300 + 42164400 = 85675700

Coalition cost = 48416500

Savings = (85675700 – 48416500) / 85675700 = 43.5 %

Circle K / Esso

Summed stand-alone costs = 43511300 + 48063600 = 91574900

Coalition cost = 48798900

Savings = (91574900 – 48798900) / 91574900 = 46.7 %

Circle K / YX

Summed stand-alone costs = 43511300 + 73247100 = 116758400

Coalition cost = 44937700

Savings = (116758400 – 44937700) / 116758400 = 61.5 %

Shell / Esso

Summed stand-alone costs = 42164400 + 48063600 = 90228000

Coalition cost = 31220400

Savings = (90228000 – 31220400) / 90228000 = 65.4 %

Shell / YX

Summed stand-alone costs = 42164400 + 73247100 = 115411500

Coalition cost = 57276400

Savings = (115411500 – 57276400) / 115411500 = 50.4 %

Esso / YX

Summed stand-alone costs = 48063600 + 73247100 = 121310700

Coalition cost = 51858900

Savings = (121310700 – 51858900) / 121310700 = 57.3 %

Circle K / Shell / Esso

Summed stand-alone costs = 43511300 + 42164400 + 48063600 = 133739300

Coalition cost = 31895200

Savings = (133739300 – 31895200) / 133739300 = 76.2 %

Circle K / Shell / YX

Summed stand-alone costs = 43511300 + 42164400 + 73247100 = 158922800

Coalition cost = 56136300

Savings = (158922800 – 56136300) / 158922800 = 64.7 %

Circle K / Esso / YX

Summed stand-alone costs = 43511300 + 48063600 + 73247100 = 164822000

Coalition cost = 36930300

Savings = (164822000 – 36930300) / 164822000 = 77.6 %

Shell / Esso / YX

Summed stand-alone costs = 42164400 + 48063600 + 73247100 = 163475100

Coalition cost = 36930300

Savings = (163475100 – 36930300) / 163475100 = 77.4 %

Circle K / Shell / Esso / YX

Summed stand-alone costs = 43511300 + 42164400 + 48063600 + 73247100 = 206986400

Coalition cost = 34398700

Savings = 206986400 NOK – 34398700 NOK = 172587700 NOK

Savings = (206986400 – 34398700) / 206986400 = 83.4 %

Circle K / Shell / Esso / YX / Equinor depot

Summed stand-alone costs = 43511300 + 42164400 + 48063600 + 73247100 = 206986400

Coalition cost = 34398700

Savings = 206986400 NOK – 34398700 NOK = 172587700 NOK

Savings = (206986400 – 34398700) / 206986400 = 83.4 %

We see that the result here is exactly the same as when Equinor Mongstad depot is not included. This make sense as there is always another depot that are closer to any gas station than Mongstad and now all depots have limitless capacity. So Mongstad will now never deliver fuel to any gas station and we got therefore equal results as in the previous collaboration.

5.2.3 Shapley values

Circle K: 7337550

Shell: 5406150

Esso: 740500

YX: 20914500

5.2.4 Individual rational

Circle K: 7337550 < 43511300 (yes)

Shell: 5406150 < 42164400 (yes)

Esso: 740500 < 48063600 (yes)

YX: 20914500 < 73247100 (yes)

5.2.5 Coalition rational

Circle K / Shell: (7337550 + 5406150) < 48416500 (yes)

Circle K / Esso: (7337550 + 740500) < 48798900 (yes)

Circle K / YX: (7337550 + 20914500) < 44937700 (yes)

Shell / Esso: (5406150 + 740500) < 31220400 (yes)

Shell / YX: (5406150 + 20914500) < 57276400 (yes)

Esso / YX: (740500 + 20914500) < 51858900 (yes)

Circle K / Shell / Esso: (7337550 + 5406150 + 740500) < 31895200 (yes)

Circle K / Shell / YX: (7337550 + 5406150 + 20914500) < 56136300 (yes)

Circle K / Esso / YX: (7337550 + 740500 + 20914500) < 38757700 (yes)

Shell / Esso / YX: (5406150 + 740500 + 20914500) < 36930300 (yes)

Since I only have yeses, the rational condition for this cost allocation are fulfilled.

5.2.6 Efficiency condition

Circle K / Shell / Esso / YX: (7337550 + 5406150 + 740500 + 20914500) = 34398700 (yes)

5.2.7 Conclusion from the individual rational-, coalition rational- and efficiency condition tests

This cost allocation from the Shapley values formula passed all three tests, which means according to game theory, that this cost allocation is stable and belongs to the core.

5.2.8 Savings for each company

Circle K:

43511300 - 7337550 = 36173750

Annually savings = 36173750 NOK

36173750 / 43511300 = 0.831

Percentage savings = 83.1 %

Shell:

42164400 - 5406150 = 36758250

Annually savings = 36758250 NOK

36758250 / 42164400 = 0.872

Percentage savings = 87.2 %

Esso:

48063600 - 740500 = 47323100

Annually savings = <u>47323100 NOK</u>

47323100 / 48063600 = 0.985

Percentage savings = <u>98.5 %</u>

YX:

73247100 - 20914500 = 52332600

Annually savings = <u>52332600 NOK</u>

52332600 / 73247100 = 0.714

Percentage savings = <u>71.4 %</u>

5.2.9 Reduction in CO^2 emission at full collaboration

2660 gram x 0.01 = 26.6 gram

From this gives us that the trucks emission of CO^2 is 26.6 gram per 1000 litre km of distributed fuel.

26.6 gram = 0.0000266 ton

0.0000266 / 0.31 = 0.00008581

0.00008581 x 172587700 = 14809.8

Annual reduction in CO^2 emission from full collaboration with limitless depots = <u>14809.8 ton</u>

5.3 Fuel distribution with adjusted depot capacities

Here I have run a new series of 16 analyses with total depot capacities adjusted to relate to each company's share of annual fuel sale at all their gas stations in Norway. All values are then multiplied with 20% to make some overcapacity. The total overcapacity will in addition to this also include the depot at Mongstad. For Equinor which do not own or operate any gas stations I have calculated an average capacity from the 4 companies for their depot capacity at Mongstad.

Annual capacities in 1000 litre:

Circle K:	4874893 x 1.2 x 0,325/ 6 = 316868.0
Shell:	4874893 x 1.2 x 0,325 / 8 = 237651.0
Esso:	4874893 x 1.2 x 0,225 / 4 = 329055.3
YX:	4874893 x 1.2 x 0,125 / 2 = 365617.0
Equinor:	(316868.0 + 237651.0 + 329055.3 + 365617.0) / 4 = 312297.8

5.3.1 Coalition transportation cost results from AMPL listed

Circle K	177207000
Shell	285499000
Esso	50214100
YX	94770200
Circle K / Shell	431180000
Circle K / Esso	129120000
Circle K / YX	226464000
Shell / Esso	196868000

Shell / YX	341711000
Esso / YX	87572400
Circle K / Shell / Esso	332446000
Circle K / Shell / YX	485696000
Circle K / Esso / YX	173934000
Shell / Esso / YX	256191000
Circle K / Shell / Esso / YX	387685000
Circle K / Shell / Esso / YX / Equinor depot	299921000

5.3.2 Savings for each coalition

Circle K / Shell

Summed stand-alone costs = 177207000 + 285499000 = 462706000

Coalition cost = 431180000

Savings = (462706000 – 431180000) / 462706000 = 6.8 %

Circle K / Esso

Summed stand-alone costs = 177207000 + 50214100 = 227421100

Coalition cost = 129120000

Savings = (227421100 – 129120000) / 227421100 = 43.2 %

Circle K / YX

Summed stand-alone costs = 177207000 + 94770200 = 271977200

Coalition cost = 226464000

Savings = (271977200 – 226464000) / 271977200 = 16.7 %

Shell / Esso

Summed stand-alone costs = 285499000 + 50214100 = 335713100

Coalition cost = 196868000

Savings = (335713100 – 196868000) / 335713100 = 42.4 %

Shell / YX

Summed stand-alone costs = 285499000 + 94770200 = 380269200

Coalition cost = 341711000

Savings = (380269200 – 341711000) / 380269200 = 10.1 %

Esso / YX

Summed stand-alone costs = 50214100 + 94770200 = 144984300

Coalition cost = 87572400

Savings = (144984300 – 87572400) / 144984300 = 39.6 %

Circle K / Shell / Esso

Summed stand-alone costs = 177207000 + 285499000 + 50214100 = 512920100

Coalition cost = 332446000

Savings = (512920100 – 332446000) / 512920100 = 35.2 %

Circle K / Shell / YX

Summed stand-alone costs = 177207000 + 285499000 + 94770200 = 557476200

Coalition cost = 485696000

Savings = (557476200 – 485696000) / 557476200 = 12.9 %

Circle K / Esso / YX

Summed stand-alone costs = 177207000 + 50214100 + 94770200 = 322191300

Coalition cost = 173934000

Savings = (322191300 – 173934000) / 322191300 = 46.0 %

Shell / Esso / YX

Summed stand-alone costs = 285499000 + 50214100 + 94770200 = 430483300

Coalition cost = 256191000

Savings = (430483300 – 256191000) / 430483300 = 40.5 %

Circle K / Shell / Esso / YX

Summed stand-alone costs = 177207000 + 285499000 + 50214100 + 94770200 = 607690300

Coalition cost = 387685000

Savings = 607690300 NOK – 387685000 NOK = 220005300 NOK

Savings = (607690300 – 387685000) / 607690300 = 36.2 %

Circle K / Shell / Esso / YX / Equinor depot

Summed stand-alone costs = 177207000 + 285499000 + 50214100 + 94770200 = 607690300

Coalition cost = 299921000

Savings = 607690300 NOK – 299921000 NOK = 307769300 NOK

Savings = (607690300 – 299921000) / 607690300 = 50.6 %

We see that the result for Circle K / Shell / Esso / YX / Equinor depot collaboration now is different than when Equinor Mongstad depot was not included. In the case where we had unlimited supply, Equinor's depot had no effect on optimal solution. Now Equinor's depot has an effect on optimal solution due to the fact that maximum supply from the fuel depots has changed, in this case where we have "adjusted depot capacities.

5.3.3 Shapley values

Circle K: 137359025.0

Shell: 231374525.0

Esso: -43674208.3

YX: 62625658.3

5.3.4 Individual rational

Circle K: 137359025.0 < 177207000 (yes)

Shell: 231374525.0 < 285499000 (yes)

Esso: -43674208.3 < 50214100 (yes)

YX: 62625658.3 < 94770200 (yes)

5.3.5 Coalition rational

Circle K / Shell: (137359025 + 231374525) < 431180000 (yes)

Circle K / Esso: (137359025 +(-43674208.3)) < 129120000 (yes)

Circle K / YX: (137359025 + 62625658.3) < 226464000 (yes)

Shell / Esso: (231374525 +(-43674208.3)) < 196868000 (yes)

Shell / YX: (231374525 + 62625658.3) < 341711000 (yes)

Esso / YX: ((-43674208,3) + 62625658.3) < 87572400 (yes)

Circle K / Shell / Esso: (137359025 + 231374525 + (-43674208.3)) < 332446000 (yes)

Circle K / Shell / YX: (137359025 + 231374525 + 62625658.3) < 485696000 (yes)

Circle K / Esso / YX: (137359025 + (-43674208.3) + 62625658.3) < 173934000 (yes)

Shell / Esso / YX: (231374525 + (-43674208.3) + 62625658.3) < 256191000 (yes)

Since I only have yeses, the rational condition for this cost allocation is fulfilled.

5.3.6 Efficiency condition

Circle K / Shell / Esso / YX: (137359025 + 231374525 + (-43674208.3)) + 62625658.3 = 387685000 (yes)

The sum for all allocated costs are equal to the full collaboration coalition cost from AMPL, I then conclude that the efficient condition is fulfilled.

5.3.7 Conclusion from the individual rational-, coalition rational-, and efficiency condition tests

This cost allocation from the Shapley values formula passed all three tests, which means according to game theory, that this cost allocation is stable and belongs to the core.

5.3.8 Savings for each company

Circle K:

177207000 - 137359025 = 40047975

Annually savings = 40047975 NOK

40047975 / 177207000 = 0.226

Percentage savings = 22.6 %

Shell:

285499000 – 231374525 = 54124475

Annually savings = 54124475 NOK

54124475 / 285499000 = 0.190

Percentage savings = 19.0 %

Esso:

50214100.0 – (-43674208.3) = 93888308.3

Annually savings = 93888308.3 NOK

93888308.3 / 50214100 = 1.870

Percentage savings = 187.0 %

YX:

94770200 – 62625658.3 = 32144541.7

Annually savings = 32144541.7 NOK

32144541.7 / 94770200 = 0.339

Percentage savings = 33.9 %

5.3.9 Reduction in CO^2 emission at full collaboration

2660 gram x 0.01 = 26.6 gram

From this gives us that the trucks emission of CO^2 is 26.6 gram per 1000 litre km of distributed fuel.

26.6 gram = 0.0000266 ton

0,0000266 / 0.31 = 0.00008581

0.00008581 x 307769300 = 26409.7

Annual reduction in CO^2 emission from full collaboration with adjusted depot capacities = 26409.7 ton

6. Discussion

I have run 3 series of 15 analyses and from the results used Shapley Values to allocate the grand coalition cost to the 4 companies Circle K, Shell, Esso and YX/UNO-X. I then checked each company's allocated cost from Shapley Values regarding the 2 conditions individual- and rational condition, to find out if the cost allocation is stable and belongs to the core. I also checked the sum regarding efficiency condition. All this is done 3 times(3 series) due to 3 different games.

The allocated costs passed all these test in every 3 series. So we then know that the cost allocations in every 3 cases are stable and belongs to the core. This means that all the 4 companies Circle K, Shell, Esso and YX/UNO-X have most benefit from stay in the grand coalition, given the input data and restrictions in these 3 series of AMPL analyses. As all three series of analyses and calculations show that the grand coalition is best for all the four companies, I suggest that this probably is the case in reality as well. The reason I believe that this likely can be true is that I believe the true depot capacities probably lay within the area of these 3 series of capacities I have used in the analyses, I know capacity may differ within each depot in reality, but at least I still think this indicates that these results can be true, in a way that all 4 companies will have most benefit regarding cost from collaborating in the 4 company coalition where they are open to share depots in a optimized fuel distribution, and that this also of course gives the greatest reduction in CO^2 emission as this will have the same percentage reduction as the percentage savings.

The results show in all three series that every of the 4 companies have considerable savings from the optimized collaborated fuel distributing in this grand coalition of all 4 companies collaborating. The results regarding decreased CO^2 emission also shows considerable amounts decreased CO^2 emissions in all 3 series.

However, I believe that the results for percentage- savings and decreased CO^2 emission are more likely to be closer to the reality then the savings and decreased CO^2 emission in NOK. This is due to some uncertainty regarding cost from driving and salary, as changes in this cost will affect the savings in NOK, but not in percentage.

As I explained earlier the return routes are not included in the program, so the savings will in practice probably be higher than the results here, not in percent, but in NOK. The amount of decreased CO^2 emission will of course also probably be higher in practice, due to this.

The results for total savings from full collaborated (included Equinor's depot) fuel distributing in Norway compared to non-collaborative distribution, from the 3 series analyses, given the input data show:

Annual total savings in NOK (serie 1) = 196145400 NOK

Annual total savings in percent (serie 1) = 48.7 %

Annual total decreased CO^2 emission in ton (serie 1) = 16830.54 ton

Annual total savings in NOK (serie 2) = 172587700 NOK

Annual total savings in percent (serie 2) = 83.4 %

Annual total decreased CO^2 emission in ton (serie 2) = 14809.8 ton

Annual total savings in NOK (serie 3) = 307769300 NOK

Annual total savings in percent (serie 3) = 50.6 %

Annual total decreased CO^2 emission in ton (serie 3) = 26409.7 ton

From the main results we can see that the greatest savings in percent is for the analyses with "limitless depot capacities"(serie 2), but still have less savings in NOK then the results for "equal depot capacities"(serie 1). This make sense as the companies now can distribute all the fuel from the nearest depot no matter how much fuel they want, this makes low costs. At the same time they will have high savings when they can pic up as much fuel as they want from all any of the 21 depots in Norway.

If we takes a look at the results in the results chapter there also seems to be a pattern that Esso have the greatest contribution to the savings among the companies and that this contribution is very high compared to the contribution from the other companies. We can see that the coalitions where Esso is included tend to have high percentage savings. Though all companies still contributes significantly to the savings. The reason for Essos high contribution to the savings can be that Esso have depots which are more suitable spread in the country regarding locations and demands, compared to the other companies.

We see that Esso do have negative allocated cost value from Shapley Values. This means that when Circle K, Shell and YX have paid their allocated costs into the money pot for the grand coalition cost Esso do not pay anything to the money pot but rather gain a sum from that money pot. This is due to Esso’s very high contribution to the savings in the collaboration. Esso’s contribution is as high that the cost for the grand coalition is way lower than the cost would be for a 3 companies coalition where Esso would not be included. The average marginal cost of Esso is negative. The grand coalition cost is actual lower than the sum for what Circle K, Shell and YX pays into the grand coalition pot, due to that Esso get some of this money. This cost allocation is still stable and belongs to the core, all companies are most beneficial to stay in the grand coalition, they has no reason to break out to form smaller coalitions. However, the companies do not necessarily need to split the grand coalition cost exactly this way, they can agree to split it different, but the cost allocation from Shapley Values here is probably a good suggestion to how the companies should split the grand coalition cost in a fair way, even though Esso get paid from the money pot.

The results for the savings from 2- and 3 company coalitions are actual not very important here as we already know that the grand coalition gives most benefits for all the companies. But these percentage saving results can sure shred more light on the results and show some patterns which gives a better understanding of the results, I will therefore discuss some of these findings.

We can see that there tend to be more savings as larger the coalitions are, that make sense as collaboration here is most beneficial in the grand coalition, where all four companies collaborates.

Shell seems to have the lowest contribution to the savings, but do not differ very much from Circle K and YX. One reason for this can be that they have

We see that a 3 company coalition with Circle K, Esso and YX have the highest percentage saving in 2 series, but all these 3 companies are still more beneficial when they including Shell, because Shells allocated cost in the grand coalition will be high enough to lower the other companies cost, which gives the other companies higher benefits in the grand coalition then in a 3 company coalition, and Shell still contributes enough to the savings to get an allocated cost which is lower than its own stand alone cost. Every company will still be most beneficial to stay in the grand coalition where all 4 companies collaborating, but that did we already know according to the 3 condition tests. Each company contributes to savings in the grand coalition, but with different amounts and at different grades. The companies will get allocated their costs based on how much they contributes to the savings.

The analyses and calculations for decreased CO^2 emissions show that the amount of annual decreased CO^2 emission from a full collaboration is considerable.

Although the results in this thesis of course is not perfectly accurate to reality, due particularly to uncertain depot capacities, and changes in depot capacities can sure lead to changed numbers in the results, but I still believe that the results gives true pointers of each companies savings of being in the collaboration and how they want to collaborate. And maybe more important, I think that the model gives true pointers of the magnitude of the amount of decreased CO^2 emission collaboration potential can give in distributing fuel in Norway.

A aspect to think of when creating a program in AMPL for collaboration regarding the results for which depot one should pic up fuel from and how much, is when two depots are located very close to each other. The program then chooses only the nearest depot as long as it has capacity to deliver, no matter how minimal the difference in distances is. The coalition cost will almost not be changed if one pic up fuel also from the depot laying minimal far away from the gas station, but this changes which depot (company) should deliver the fuel to a specific city/place (gas station). So this should be pay attention to when creating the program in AMPL. Though this aspect is not important in the AMPL analyses here when the purpose is to find savings for the entire coalitions.

7. Conclusion

The results show that all the four companies are most beneficial to collaborate together in a 4 company coalition (grand coalition), given the input data and restrictions in the analyses. The results also show that the savings from a optimized fuel distribution in the grand coalition compared to a none collaborative fuel distribution are not even among the companies, but still are considerable for all the four companies.

From this I recommend all the four companies Circle K, Shell (ST1), Esso and YX/UNO-X to collaborate together in the 4 companies coalition (grand coalition), being open to share depots, in distributing fuel in Norway, to get the lowest possible transportation costs for their own company. They than need to arrange the routes and amounts as the text file in AMPL displays. Further I recommend the companies to share the costs according to the results from Shapley Values. This is given that there will be done new analyses with accurate input data to get as accurate results as possible and to verify my conclusions.

The results also show that the amount of potential decreased CO^2 emission from the trucks used to distribute the fuel in Norway is considerable when all four companies collaborate together in a 4 company coalition compared to a none collaborative fuel distribution.

From this I also recommend all the four companies Circle K, Shell (ST1), Esso and YX/UNO-X to collaborate together in the 4 companies coalition (grand coalition), being open to share depots, in distributing fuel in Norway, to reduce the CO^2 emissions to the world as much as possible. In this aspect also the government can play an important role in facilitating and encourage to collaboration in distributing fuel in Norway. They than need to arrange the routes and amounts as the text file in AMPL says. This is given that there will be done new analyses with accurate input data to get as accurate results as possible and to verify my conclusions.

Further I experiences that the AMPL program I have created in this thesis is working smooth and is easy to change and expand regarding changed or added data. The program could therefore be used as a tool or a help in decision makings in collaborative fuel distribution in Norway.

8. References

Centon, W. (2019, November 13). *Economics*. Retrieved from Behavior Economics: http://www.investopedia.com/terms/s/shapley-value.asp

Coalition Theory Network. (2020). Retrieved from Cooperative Game Theory.

Cotra, M. (2019, October 28). *Towards Datascience*. Retrieved from Making Sense of Shapley Values: https://towardsdatasciensce.com/making-sense-of-shapley-values-dc67a8e4c5e8

GlobalPetrolPrices. (2020, October 27). *Dieselpriser, Liter, 27-oktober-2020*. Retrieved from GlobalPetrolPrices.com: https://no.globalpetrolprices.comdiesel.prices/#hl214

Google. (2005, February 8). *Google*. Retrieved from Google Maps: https://www.google.no/maps/dir///@60.5821454,5.7955337,14z?hl=en

Havås, J., Alfred, O., Jim, P., & Mirjam, S. S. (2013). *Modeling and Optimization of University Timetabling.* Gøteborg: Gøteborgs Universitet.

Helleborg, B. (2018, May 29). *Om verden og oss som bor her*. Retrieved from Hvor mye drivhusgass slippes ut nå vi kjører bil: https://www.om-verden.com/hvor-mye-drivhusgass-slippes-ut-na-vi-kjorer-bil/

Johansen, K. E. (2010). *Det norske drivstoffmarkedet.* Bergen: Norwegian Competition Authority.

Robert Fourer, D. M. (2003). *AMPL Streamlined Modeling for Real Optimization*. Retrieved from AMPL: A Modeling Language for Mathematical Programming: https://ampl.com/resources/the-ampl-book/chapter-downloads/

Spilde, D., & Skotland, C. (2016). *Hvordan vil en omfattende elektrifisering av transportsektoren påvirke kraftsystemet?* NVE.

Statistisk Sentralbyrå. (2020, April 20). *Statistisk Sentralbyrå*. Retrieved from Sales of Petroleumproducts: https://www.ssb.no/en/energi-og-industri/statistikker/petroleumsalg/aar

Thorsnæs, G. (2019, September 29). *Store norske leksikon*. Retrieved from De største byene i Norge: https://snl.no/de_største_byene_i_Norge

Appendix: AMPL file names

Equal depot capacities

Model file

Model.mod:	The model file used for all analyses

Data files

DataCi.dat:	Circle K
DataSh.dat:	Shell
DataEs.dat:	Esso
DataYX.dat:	YX
DataCiSh.dat:	Circle K / Shell
DataCiEs.dat:	Circle K / Esso
DataCiYX.dat:	Circle K / YX
DataShEs.dat:	Shell / Esso
DataShYX.dat:	Shell / YX
DataEsYX.dat:	Esso / YX
DataCiShEs.dat:	Circle K / Shell / Esso
DataCiShYX.dat:	Circle K / Shell / YX
DataCiEsYX.dat:	Circle K / Esso / YX
DataShEsYX.dat:	Shell / Esso / YX
DataCiShEsYX.dat:	Circle K / Shell / Esso / YX
Data.dat:	Circle K / Shell / Esso / YX / Equinor depot

Run files

RunCi.run:	Circle K
RunSh.run:	Shell
RunEs.run:	Esso
RunYX.run:	YX
RunCiSh.run:	Circle K / Shell
RunCiEs.run:	Circle K / Esso
RunCiYX.run:	Circle K / YX
RunShEs.run:	Shell / Esso
RunShYX.run :	Shell / YX
RunEsYX.run:	Esso / YX
RunCiShEs.run:	Circle K / Shell / Esso
RunCiShYX.run:	Circle K / Shell / YX
RunCiEsYX.run:	Circle K / Esso / YX
RunShEsYX.run:	Shell / Esso / YX
RunCiShEsYX.run:	Circle K / Shell / Esso / YX
Run.run:	Circle K / Shell / Esso / YX / Equinor depot

Text files

TextCi.txt:	Circle K
TextSh.txt:	Shell
TextEs.txt:	Esso
TextYX.txt:	YX
TextCiSh.txt:	Circle K / Shell
TextCiEs.txt:	Circle K / Esso
TextCiYX.txt:	Circle K / YX
TextShEs.txt:	Shell / Esso
TextShYX.txt :	Shell / YX
TextEsYX.txt:	Esso / YX
TextCiShEs.txt:	Circle K / Shell / Esso
TextCiShYX.txt:	Circle K / Shell / YX
TextCiEsYX.txt:	Circle K / Esso / YX
TextShEsYX.txt:	Shell / Esso / YX
TextCiShEsYX.txt:	Circle K / Shell / Esso / YX
Text.txt:	Circle K / Shell / Esso / YX / Equinor depot

Limitless depot capacities

Model file

Model.mod:	The model file used for all analyses

Data files

DataLICi.dat:	Circle K
DataLISh.dat:	Shell
DataLIEs.dat:	Esso
DataLIYX.dat:	YX
DataLICiSh.dat:	Circle K / Shell
DataLICiEs.dat:	Circle K / Esso
DataLICiYX.dat:	Circle K / YX
DataLIShEs.dat:	Shell / Esso
DataLIShYX.dat:	Shell / YX
DataLIEsYX.dat:	Esso / YX
DataLICiShEs.dat:	Circle K / Shell / Esso
DataLICiShYX.dat:	Circle K / Shell / YX
DataLICiEsYX.dat:	Circle K / Esso / YX
DataLIShEsYX.dat:	Shell / Esso / YX
DataLICiShEsYX.dat:	Circle K / Shell / Esso / YX
DataLI.dat:	Circle K / Shell / Esso / YX / Equinor depot

Run files

RunLICi.run:	Circle K
RunLISh.run:	Shell
RunLIEs.run:	Esso
RunLIYX.run:	YX
RunLICiSh.run:	Circle K / Shell
RunLICiEs.run:	Circle K / Esso
RunLICiYX.run:	Circle K / YX
RunLIShEs.run:	Shell / Esso
RunLIShYX.run:	Shell / YX
RunLIEsYX.run:	Esso / YX
RunLICiShEs.run:	Circle K / Shell / Esso
RunLICiShYX.run:	Circle K / Shell / YX
RunLICiEsYX.run:	Circle K / Esso / YX
RunLIShEsYX.run:	Shell / Esso / YX
RunLICiShEsYX.run:	Circle K / Shell / Esso / YX
RunLI.run:	Circle K / Shell / Esso / YX / Equinor depot

Text files

TextLICi.txt:	Circle K
TextLISh.txt:	Shell
TextLIEs.txt:	Esso
TextLIYX.txt:	YX
TextLICiSh.txt:	Circle K / Shell
TextLICiEs.txt:	Circle K / Esso
TextLICiYX.txt:	Circle K / YX
TextLIShEs.txt:	Shell / Esso
TextLIShYX.txt:	Shell / YX
TextLIEsYX.txt:	Esso / YX
TextLICiShEs.txt:	Circle K / Shell / Esso
TextLICiShYX.txt:	Circle K / Shell / YX
TextLICiEsYX.txt:	Circle K / Esso / YX
TextLIShEsYX.txt:	Shell / Esso / YX
TextLICiShEsYX.txt:	Circle K / Shell / Esso / YX
TextLI.txt:	Circle K / Shell / Esso / YX / Equinor depot

Adjusted depot Capacities

Model file

Model.mod:	The model file used for all analyses

Data files

DataADCi.dat:	Circle K
DataADSh.dat:	Shell
DataADEs.dat:	Esso
DataADYX.dat:	YX
DataADCiSh.dat:	Circle K / Shell
DataADCiEs.dat:	Circle K / Esso
DataADCiYX.dat:	Circle K / YX
DataADShEs.dat:	Shell / Esso
DataASDShYX.dat:	Shell / YX
DataADEsYX.dat:	Esso / YX
DataADCiShEs.dat:	Circle K / Shell / Esso
DataADCiShYX.dat:	Circle K / Shell / YX
DataADCiEsYX.dat:	Circle K / Esso / YX
DataADShEsYX.dat:	Shell / Esso / YX
DataADCiShEsYX.dat:	Circle K / Shell / Esso / YX
DataAD.dat:	Circle K / Shell / Esso / YX / Equinor depot

Run files

RunADCi.run:	Circle K
RunADSh.run:	Shell
RunADEs.run:	Esso
RunADYX.run:	YX
RunADCiSh.run:	Circle K / Shell
RunADCiEs.run:	Circle K / Esso
RunADCiYX.run:	Circle K / YX
RunADShEs.run:	Shell / Esso
RunADShYX.run:	Shell / YX
RunADEsYX.run:	Esso / YX
RunADCiShEs.run:	Circle K / Shell / Esso
RunADCiShYX.run:	Circle K / Shell / YX
RunADCiEsYX.run:	Circle K / Esso / YX
RunADShEsYX.run:	Shell / Esso / YX
RunADCiShEsYX.run:	Circle K / Shell / Esso / YX
RunAD.run:	Circle K / Shell / Esso / YX / Equinor depot

Text files

TextADCi.txt:	Circle K
TextADSh.txt:	Shell
TextADEs.txt:	Esso
TextADYX.txt:	YX
TextADCiSh.txt:	Circle K / Shell
TextADCiEs.txt:	Circle K / Esso
TextADCiYX.txt:	Circle K / YX
TextADShEs.txt:	Shell / Esso
TextADShYX.txt:	Shell / YX
TextADEsYX.txt:	Esso / YX
TextADCiShEs.txt:	Circle K / Shell / Esso
TextADCiShYX.txt:	Circle K / Shell / YX
TextADCiEsYX.txt:	Circle K / Esso / YX
TextADShEsYX.txt:	Shell / Esso / YX
TextADCiShEsYX.txt:	Circle K / Shell / Esso / YX
TextAD.txt:	Circle K / Shell / Esso / YX / Equinor depot

Equal depot capacities CO^2 emission

Model file

Model.mod:	The model file used for all analyses

Data files

DataCOCi.dat:	Circle K
DataCOSh.dat:	Shell
DataCOEs.dat:	Esso
DataCOYX.dat:	YX
DataCO.dat:	Circle K / Shell / Esso / YX / Equinor depot

Run files

RunCOCi.run:	Circle K
RunCOSh.run:	Shell
RunCOEs.run:	Esso
RunCOYX.run:	YX
RunCO.run:	Circle K / Shell / Esso / YX / Equinor depot

Text files

TextCOCi.txt:	Circle K
TextCOSh.txt:	Shell
TextCOEs.txt:	Esso
TextCOYX.txt:	YX
TextCO.txt:	Circle K / Shell / Esso / YX / Equinor depot