NATURAL GAS NETWORK DESIGN UNDER DEMAND UNCERTAINTY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ERINÇ BARIŞ KOÇ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

DECEMBER 2019

Approval of the thesis:

NATURAL GAS NETWORK DESIGN UNDER DEMAND UNCERTAINTY

submitted by **ERINÇ BARIŞ KOÇ** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Yasemin Serin
Head of Department, **Industrial Engineering** _____

Assoc. Prof. Dr. Canan Şerbetcioğlu
Supervisor, **Industrial Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Yasemin Serin
Industrial Engineering, METU _____

Assoc. Prof. Dr. Canan Şerbetcioğlu
Industrial Engineering, METU _____

Assoc. Prof. Dr. Sedef Meral
Industrial Engineering, METU _____

Assist. Prof. Dr. Özgen Karaer
Industrial Engineering, METU _____

Assoc. Prof. Dr. Alp Ertem
Industrial Engineering, Çankaya University _____

Date: 20.12.2019

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Erinç Barış Koç

Signature:

ABSTRACT

NATURAL GAS NETWORK DESIGN UNDER DEMAND UNCERTAINTY

Koç, Erinç Barış
Master of Science, Industrial Engineering
Supervisor: Assoc. Prof. Dr. Canan Şerbetcioğlu

December 2019, 127 pages

Pipeline and storage network design problem is to design a pipeline network with its constituting arcs and their capacities to meet natural gas demands of the cities, while minimizing the total cost of the system. Storage nodes are included in the network structure to store the natural gas in cases of insufficient supply. Models for the deterministic and stochastic versions of the problem are designed. The deterministic model is solved with the Benders Decomposition method. Three different versions of Benders decomposition are used. For the stochastic model, two-stage stochastic programming is used. Computational experiments are performed on randomly generated problem instances. A case study for the Turkish pipeline system is also analyzed.

Keywords: Pipeline Network Design, Benders Decomposition, Stochastic Programing

ÖZ

RASSAL TALEP ALTINDA DOĞAL GAZ BORU HATTI TASARIMI

Koç, Erinç Barış
Yüksek Lisans, Endüstri Mühendisliği
Tez Danışmanı: Doç. Dr. Canan Şerbetcioğlu

Aralık 2019, 127 sayfa

Doğal gaz boru hattı ve depolama ağ tasarımı problemi, ağın doğal gaz talebini karşılaması amacı ile boru hatlarını kapasite ve talep kısıtları altında oluşturulurken toplam maliyeti en aza indirir. Doğal gaz depolama tesisleri yetersiz tedarik durumlarında kullanmak için ağın içine eklenirler. Matematiksel model deterministik ve stokastik olarak geliştirilmiştir. Deterministik model Benders ayrışımı kullanılarak çözülmüştür. Üç farklı Benders ayrışım türevi kullanılmıştır. İki kademeli stokastik optimizasyon modeli kullanılarak stokastik model çözülmüştür. Bir örnek olay olarak Türkiye boru hatları sistemi çalışılmıştır. Hesaplamalı incelemeler rasgele oluşturulmuş problemler üzerinde çalışılmıştır.

Anahtar Kelimeler: Doğal Gaz Boru Hattı Tasarımı, Benders Ayrışımı, Stokastik Programlama

To Gülhan and Orhan

ACKNOWLEDGEMENTS

First of all, I would like to express my special thanks to Assoc. Prof. Dr. Canan Sepil for her supervision. Apart from my advisor, I would like to thank to my thesis committee. In addition, I would like to thank to my friends for their support and motivation. Finally, I would like to express my appreciations to my mother Gülhan and my father Orhan for their support for all my life.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

Natural gas is a way of clean, safe and cheaper energy resource. It produces less carbon dioxide with respect to coal or petroleum so that it directly decreases the greenhouse gas effect in the world. These impacts make natural gas a more favorable choice of energy source.

The transportation of natural gas requires a sophisticated system that carries the gas from the production sites and/or entry points to the demand points. In this study, a natural gas network design problem is studied. In the other parts of this section, the basic facts about natural gas are provided. In Section 1.1, we review production and consumption patterns of natural gas. In Section 1.2, natural gas pipeline network elements are analyzed. In Section 1.3, we explain the design of natural gas network systems. In Section 1.4, the contribution of the thesis is explained.

1.1. Natural Gas Production and Consumption

The natural gas usage is highly favorite for residential, industrial and electrical sectors. Natural gas demand depends on total energy demand and supply. There is a limited number of alternatives for electricity and heating purposes in short periods. Factors on supply side are natural gas import, storage levels and net production amounts. In addition, demands depend on weather, economic conditions.

The natural gas consumption rate increases as a result of growth in population and economy. The global natural gas demand is expected to increase by 50% up to 2040 (BP, 2019). It is stated that natural gas will yield the largest production increase within fossil fuels (International Energy Outlook 2019). The natural gas production is expected to grow 7% between 2018 and 2020. After 2020, the expected growth rate will be close to 1%.

The technological developments in production side decreases cost and yields higher resource availability. It is expected to have less increase in natural gas price as a result of technological developments. The natural gas production patterns and price can be seen in Figure 1.1. The production of natural gas has increased gradually up to 2020. However, the natural gas prices show fluctuations at the same time. After 2020, there exist three different projections in the graph. In case of low oil and gas resource technology, gas price will increase sharply. The reverse movement in the gas price is expected if there exist high technological development. It is sure to happen that the production of natural gas will reach to nearly 53 trillion cubic feet in 2050. The effect of technological developments is also analyzed in the graph. The flat natural gas prices are obtained by the highest technological developments.

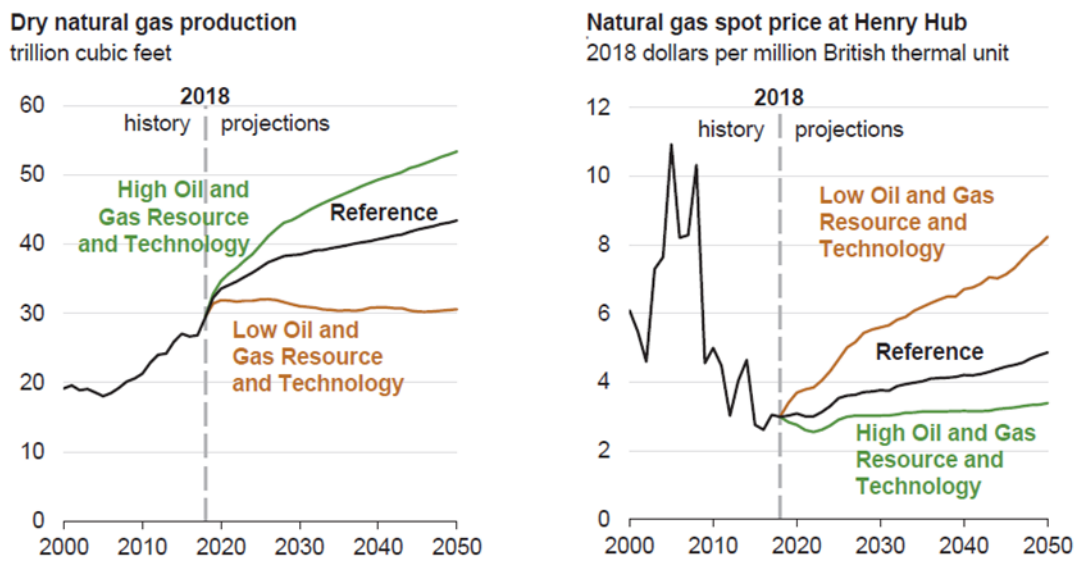

Figure 1.1. Natural Gas Production and Price Patterns [7]

The world energy consumption can be seen in Table 1.1. The important part of energy consumption is supplied from high carbon released sources. It is expected to have higher inclination towards lower-carbon energy systems. The percentage of natural gas and renewable energy sources is expected to reach 85% of the whole demand (BP 2019).

Table 1.1. *Consumption Values of Different Energy Sources in million tons of oil equivalent* [28]

	Year				
Source	2013	2014	2015	2016	2017
Coal	3865.3	3862.2	3765.0	3706.0	3731.5
Oil	4359.3	4394.7	4475.8	4557.3	4621.9
Natural Gas	2899.0	2922.3	2987.3	3073.2	3156.0
Nuclear	563.8	575.0	582.8	591.2	596.4
Hydroelectricity	859.4	879.7	880.5	913.3	918.6
Renewable	282.6	320.1	368.8	417.4	486.8

The natural gas consumption by different sectors can be seen in Figure 1.2. The industrial and electrical power sector has the highest share in the consumption. The growth rate of the industrial sector is higher than that of the electrical sector.

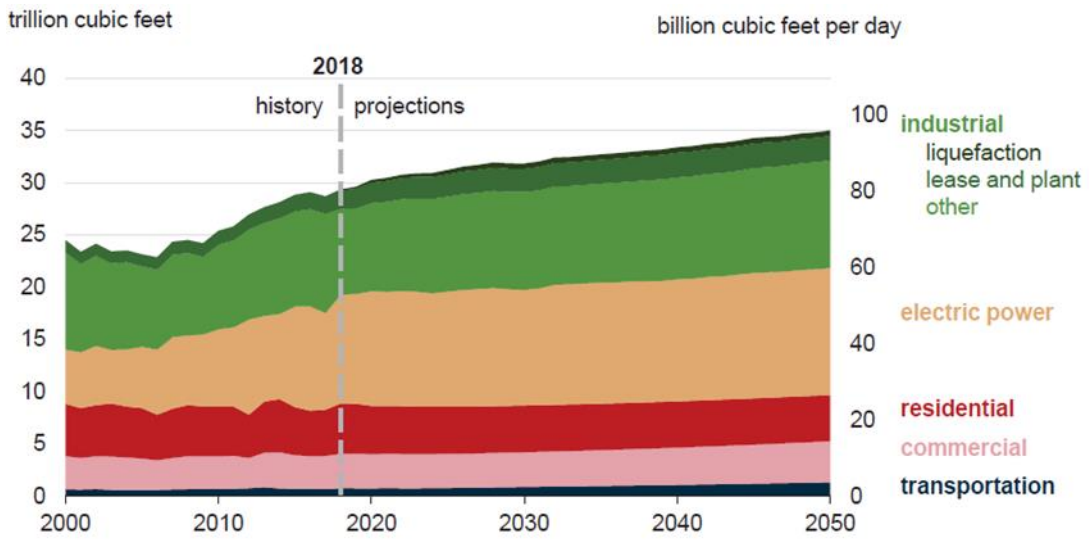

Figure 1.2. Natural Gas Consumption by Sectors [8]

Natural gas consumption data over the years 2013-2017 can be seen in Table 1.2. USA, Russia, China and Iran are taking front places in terms of demand values. Among the countries with the highest demand, China and Iran yields the highest

increase. Gas demand has shown steady rate for Saudi Arabia, Japan, Germany, UK and Italy.

Table 1.2. *Natural Gas Consumption by Countries (Million Sm^3)* [28]

	Year				
Country	2013	2014	2015	2016	2017
USA	739	750	767	773	762
Russia	466	465	445	437	471
China	167	184	192	207	238
Iran	153	172	184	188	200
Canada	112	116	115	125	134
Japan	128	130	124	125	129
Saudi Arabia	82	85	87	92	98
Germany	88	79	81	89	94
UK	78	71	72	81	79
Italy	70	62	68	71	75
Turkey	46	49	48	46	53

The highest percent of the natural gas demand is used to obtain electrical power and heating purposes. There exist five main user sectors. These are industrial, electric power, residential, commercial and transportation. Natural gas is the main source to generate fuel for heating and power. It is expected that industrial sector and electrical sector will yield the largest demand for natural gas up to 2035. Electric power sector depends on natural gas to produce electricity. Heating is the main purpose of residential sector. In the transportation sector, compressors in pipeline networks and public transportation vehicles are examples of the demand for natural gas. Natural gas production has reached 3768 billion cubic meter which is its peak in 2017. Natural gas production by countries can be seen in Figure 1.3. In 2017, Russia has increased 1/3 of its global gas production. Iran and US gas production increased again. However, the European gas production has decreased.

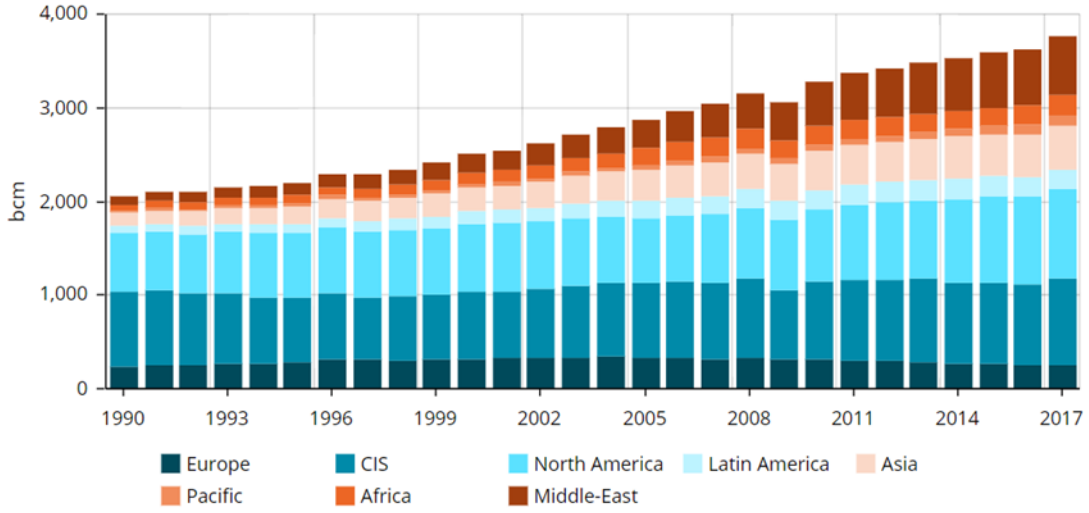

Figure 1.3. Natural Gas Production by Countries [8]

Detailed production amounts can be seen in Table 1.3. In terms of production amounts, USA, Russia, Iran and Canada generates the highest percentages.

Table 1.3. *Natural Gas Production by Countries (Million Sm^3)* [28]

	Year				
Country	2013	2014	2015	2016	2017
USA	686	733	767	759	767
Russia	675	647	638	644	694
Iran	157	175	184	190	209
Canada	156	164	165	177	184
Qatar	163	160	164	166	166
China	121	130	135	136	147
Norway	113	113	121	120	128
Australia	62	63	67	84	99
Saudi Arabia	82	85	87	92	98
Algeria	81	83	84	95	95

1.2. Pipeline Elements

There exist different types of transportation methods for natural gas. The natural gas pipeline network is the cheapest way to transfer huge amount of gas. Wang and Porcu (2008) explain that liquefied natural gas (LNG), medium conditioned liquefied gas (MLG) and compressed natural gas (CNG) are the possible transportation ways. Folga (2007) states that the pipeline network structure varies among countries. The size of the network determines the system cost and the operational parameters. The network components are pipelines, compressor stations, valves and regulators

1.2.1. Pipelines

Pipelines are generally buried underground. The sizes of pipes range between 4 inches to 48 inches depending on the usage purpose. Woldeyohannes and Majid (2011) state that the pipeline system can be divided into 3 main groups which are gathering, transmission and distribution. The gathering system collects natural gas from wells and brings it to processing facilities. Transmission systems carry natural gas between processing facilities and distribution locations. Distribution networks are used to bring natural gas to houses and industrial sides. The size, pressure limits of these pipeline systems are different from one another.

1.2.2. Compressor Stations(CS)

CS are used to increase the pressure of natural gas to transfer between locations. The maximum and minimum pressure levels should be satisfied within the pipeline system.

1.2.3. Valves and Regulators

Valves and regulators are used to adjust the pressure of the pipeline system to meet the operational limits.

1.2.4. Storage Units

The storage units are used to effectively respond to the variations in the system. Natural gas storage units are used to provide flexibility to energy systems. Especially

in high demand seasons, storage units are valuables assets to prevent shortages of gas for both residential and industrial customers. Storage units provide buffer area to accommodate the excess gas that is already paid. The storage units can be treated as warehouses. There exist different forms of natural gas storages. Natural gas can be stored as liquefied natural gas. In this form the gas volume is decreased by 600 percent. Another form of storage can be underground facilities.

1.2.5. LNG Terminals

LNG is defined as liquefied natural gas which has much smaller volume than natural gas. There exist different usages of LNG terminals in pipeline networks. LNG terminals are used to provide alternative gas sources to systems. In addition, LNG terminals can be used to store natural gas in liquefied form.

1.3. The Design of Gas Pipeline Systems

Network design problems for natural gas transportation via pipelines, the pipeline network design problem, (PNDP) deals with the determination of the network configuration of pipelines, while satisfying demand and flow constraints, with other constraints that stem from the technology. The pipeline network design problems are NP-hard problems. The mostly used methodology to design these systems is to divide the system into a number of hierarchical problems.

Corte and Sörensen (2013) state that the pipeline network design consists of 4 phases. These are:

a. Layout phase: structural design of the pipeline is obtained

b. Design phase: the pipeline characteristics such as diameter and material is obtained

c. Programming phase: generates priority order for consumers

d. Planning phase: operational plan is generated for different time periods

At the higher level of the hierarchical system, the layout phase is done and the strategic decisions about the topology of the pipeline system is determined. Here the connectivity of the network is aimed while meeting the demands and minimizing the investment costs. In the strategic level, the pipeline system network is usually formed to have a tree (or arborescence) structure, on a network consisting of supply and demand nodes.

At the tactical level, the constraints related with the flow of gas in the pipe are considered. Pressure-drop constraints determine the flow of gas in pipeline network. There are limits on the minimum and maximum pressures that can be applied. CS are used along a pipeline, generally from every 40 miles to 100 miles, to adjust the pressures of the gas. The tactical design of the pipeline system configures the optimal number of CSs, lengths and diameters of pipeline network system between the CSs and pressures at each of the CSs.

1.4. Contribution of the Thesis

In this thesis, the strategic level network design problem of gas pipelines enhanced with gas storage unit location determination is considered. The primary contribution of the study is the combined viewpoint of pipeline network design and storage unit location determination problems. The location of storage units is included in the design to cope with demand uncertainties.

In Chapter 2, the literature on strategic pipeline network design, tactical pipeline network design and the literature on gas storages are reported. The literature on the solution methods applied to network design problems are also reported in this chapter. In Chapter 3, two mathematical models are developed for the pipeline network design problem, one of them being a deterministic mathematical model and the other a stochastic mathematical model. The solution procedures for these models are given in this chapter. In Chapter 4, computational experimentation on the randomly generated problem instances is reported. In Chapter 5 the Turkish pipeline network

design problem is analyzed and reported. Chapter 6 finalizes the thesis with the conclusion.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we consider the literature on different problems related with the thesis. First some literature on network optimization problems on network design are analyzed. Then literature on strategic gas pipeline network design, tactical gas pipeline network design is given. We review studies that consider gas storage decisions, benders decomposition method and stochastic programming. We provide the mathematical model of the minimum spanning tree problem, capacity constrained minimum spanning tree problem and one terminal Telpak problem in Section 2.1. In Section 2.2, literature on strategical pipeline network problem are reported. In Section 2.3, we present the studies related with tactical gas pipeline network design problems. In Section 2.4, we review the papers that consider natural gas storage units. In Section 2.5, stochastic network design problems are analyzed. In Section 2.6 and 2.7, benders decomposition and its extension are presented. In Section 2.8, stochastic programming mathematical model structure is reviewed. In this section, two stage stochastic programing and L-shaped model are stated.

2.1. Network Optimization Problem in Network Design

Here we give some basic network optimization problems in network design.

2.1.1. Minimum Spanning Tree Problem

The minimum spanning tree problem is described on a network $G = (V,A)$ where $V = \{0, 1,2,..,N\}$ is the set of nodes (vertices) , with node 0 being the root node, and A represents the set of arcs between nodes. The cost of linking nodes i and j by an arc $(i, j)$is described by c_{ij} where $c_{ij} \geq 0$ and $c_{ij} = c_{ji}$. Gavish (1983) presented a formulation for the minimum spanning tree problem, which is based on flow of a

single commodity through the network. The formulation given by Gavish (1983) is as follows:

$$Minimize \sum_{i=1}^{N} \sum_{j=0\ j\neq i}^{N} c_{ij} y_{ij} \tag{2.1}$$

$subject\ to$

$$\sum_{j=0\ i\neq j}^{N} y_{ij} = 1 \qquad i = 1, \dots, N \tag{2.2}$$

$$\sum_{j=0\ i\neq j}^{N} x_{ij} - \sum_{j=1\ i\neq j}^{N} x_{ji} = 1 \qquad i = 1, \dots, N \tag{2.3}$$

$$x_{ij} \leq (N-1) y_{ij} \qquad (i,j) \in A \tag{2.4}$$

$$x_{ij} \geq 0 \qquad (i,j) \in A\,, \tag{2.5}$$

$$y_{ij} \in \{0,1\} \qquad (i,j) \in A\,, \tag{2.6}$$

where the decision variables are

$$y_{ij} = \begin{cases} 1, if\ arc\ (i,j) is\ used\ in\ the\ spanning\ tree \\ 0,\ \ otherwise \end{cases}$$

x_{ij} =flow on arc (i,j)

Objective function minimizes the total establishment cost of the spanning tree. Equations (2.2) express that only one arc will go into each node. Equations (2.3) provide flow balance equations and construct the tree structure. Equations (2.4) determine the arcs on the spanning tree. Equations (2.5) force variables to be positive. Equations (2.6) select binary values for variables.

2.1.2. Capacity Constrained Minimum Spanning Tree Problem (CMST)

The optimal solution of a network design problem using incapacitated minimum spanning tree may include some arcs with a large flow while some other arcs with not so large flows. Moreover, the arcs may have capacities by definition. These considerations are handled by adding capacity constraints to the minimum spanning tree problem formulation. Different formulations are developed depending on whether the demands of nodes are unitary or non-unitary.

2.1.3. One Terminal Telpak Problem

Rothfarb and Goldstein (1971) define the One-terminal Telpak Problem as a network design problem in which there is a center (or a root node) and several nodes are to be connected to the center with a minimum cost network that meets the requirements of the nodes, and the cost of arcs depend on the traffic flow on the arc. The mathematical model for the One-terminal Telpak problem is stated as follows:

$$Minimize \sum_{i=1}^{N} \sum_{j=0\ i\neq j}^{N} f_{ij} y_{ij} + \sum_{i=1}^{N} \sum_{j=0\ i\neq j}^{N} F_{ij}(x_{ij}) \tag{2.7}$$

$subject\ to$

$$\sum_{j=0\ i\neq j}^{N} y_{ij} = 1 \qquad i = 2, \dots, N \tag{2.8}$$

$$\sum_{j=0\ i\neq j}^{N} x_{ij} - \sum_{j=1\ i\neq j}^{N} x_{ji} = d_i \qquad i = 2, \dots, N \tag{2.9}$$

$$x_{ij} \leq q_{ij} y_{ij} \qquad (i,j) \in A, \tag{2.10}$$

$$x_{i1} \leq Q y_{i1} \qquad i = 1, \dots, N \tag{2.11}$$

$$x_{ij} \geq 0 \qquad (i,j) \in A \tag{2.12}$$

$$y_{ij} \in \{0,1\} \qquad (i,j) \in A, \qquad (2.13)$$

where;

d_i = traffic demand of node i, $d_i > 0, for\ \forall i \in V$

$F_{ij}(x_{ij})$ = cost of line construction which connect nodes i and j and has a traffic flow of x_{ij}

q_{ij} = capacity of arc (i,j)

f_{ij} =fixed cost of line between nodes i and j

$y_{ij} = \begin{cases} 1, if\ link\ (i,j) is\ constructed \\ 0,\ \ otherwise \end{cases}$

x_{ij} =flow amount between nodes i and j

The objective function minimizes the fixed costs of line establishment and the line configuration costs. Equations (2.8) establish a link for a node to connect other nodes. Equations (2.9) provide flow conservation. Constraints (2.10) and (2.11) ensure flow capacity control and the establishment of arcs. Constraints (2.12) force variables to be positive. Constraints (2.13) select binary values for variables.

2.2. Strategic Gas Pipeline Network Design

The pioneering work of Rothfarb et al. (1970) studies the optimal design of offshore natural gas pipeline systems. In this work, given the gas-field locations and delivery requirements, an optimal pipeline system is designed, and the optimal diameters of the pipes are determined. Non-linear pressure drop constraints are included. A heuristic procedure is applied on a spanning tree structure, for which the optimal diameters are calculated, and local improvements are performed.

Brimberg et al (2003) consider the design of pipeline network for South Gabon offshore oil field to carry from platforms to port. The configuration of the network and sizes of the pipes are determined while minimizing the investment costs. Variable neighborhood method and Tabu search is used to solve the mixed integer model heuristically. The mathematical model determines the strategical design of the pipeline. The model is tested with real life data of South Gabon Oil field.

Zhang et al. (2017) present a MIP model of offshore oil collection network in order to minimize the construction costs. In the presented model, submarine obstacles and production techniques define the constraints. The network model does not consider the pressure equations related with flow terms and includes five different discrete diameter values for the pipes. Gurobi Optimizer applied on MATLAB is used to optimize the model.

2.3. Tactical Gas Pipeline Network Design

In the tactical design problems, the natural gas pipeline network is modeled by combining continuity equations, motion equations and Bernoulli equations. Continuity equations describe the mass conservation in fluid movements. Motion equations are used to model the relationship between velocity, density and pressure of moving fluid. Bernoulli equations are used to model fluids or gases that is moving between two points. In natural gas pipelines, the governing constraints of pressure, length and diameter are nonlinear.

Mercado and Sanchez (2015) propose a detailed research work focusing on the technical parts of pipeline network. Their main effort is to focus on decision making process in pipeline network optimization. The study provides a comprehensive literature review on the gas pipeline network based on operational research aspects.

Shiono and Suziki (2016) discuss the planning process for natural gas pipeline system. The pipeline network design models determine the pipe sizes and length. The natural gas pressure is a direct function of pipeline length and diameter. The friction in pipe decrease the gas pressure as moving along pipeline. In addition, gas pressure decreases as pipe diameter decreases.

Edgar and Himmelblau (1978) develop a model to optimize the design of a gas transmission network. The model is applied on two different networks structures. Each compressor station is represented by a node and each pipeline segment by an arc between nodes. The objective function includes yearly operating and maintenance costs of compressors plus the sum of discounted capital costs of the pipelines The

solution techniques are generalized reduced gradient method for the no fixed cost case. In addition, branch and bound technique combined with nonlinear programing technique is applied for the fixed cost case. The model determines the flow rates in terms of upper and lower bounds at the nodes except the initial one. Natural gas comes only from the initial node so that one directional flow is applied. By using natural gas flows, pressure at the terminals are kept at specified levels. Moreover, the fuel consumption of natural gas is also specified in terms of 0.5% of total incoming gas. However, fuel consumption at compressor depends on not only incoming flow rates but also suction and discharge pressures.

Tabkhi et al. (2009) study high pressure gas pipeline optimization. The model determines the design properties of pipeline and necessary compressor stations based on supply and storage capabilities. The considered objective function is the total annualized cost, including the investment and operating costs. There exists multiple supply, storage and interconnection points in the pipeline network. The binary variables used to represent the flow direction of pipelines lead to a MINLP problem, solved by using the GAMS. The study includes detailed modeling formulation of each part of pipeline system. The model satisfies maximum allowable operating pressures of pipelines. Gas consumption of compressor depends on amount of compressed gas, the discharge and suction pressures.

Andre et al. (2013) develop a local search method to optimize the design of a hydrogen transmission and distribution network. The method determines the least cost topology and optimal pipeline dimensions. The model includes nonlinear pressure flow equations and uses continues diameter values. The minimum spanning tree-based approach firstly determines the initial topology. After fixing the network arcs, optimal diameter values are selected. Delta-change and Tabu Search heuristics are also applied.

Romo et al. (2009) develop a mixed integer programming model to optimize the network configuration and routing for the main Norwegian shipper of natural gas. The

main aim is to generate optimum network configuration for the main pipeline network. The work actually determines transport planning and security of supply. The model includes pressure calculations. The nonlinear pressure equations are linearized by using Taylor series expansion around a pair of input and output pressure values.

Demissie (2015) proposes a multi objective optimization model for natural gas transmission lines. The objective functions of model are fuel cost minimization and throughput maximization. The designed network structure consists of different configurations such as linear, branch and looped. Network structures consist of small number of nodes at most 5 demand nodes and 3 compressor stations. The main aim is to analyze how to operate efficiently in the given network configuration.

2.4. Natural Gas Storages

Storage units provides buffer area to accommodate the excess gas that is already paid. The operational aspect of storage units is modeled in great detail.

Gabriel et al. (2005) Egging et al. (2007), Egging and Gabriel (2006), Lise and Hobbs (2008), Zhuang (2005) focus on market equilibrium models for natural gas market by including producer, storage and pipeline operator, marketer and consumer. Storage systems buy the gas in low demand periods and store it at facilities. Stored gas is sold to end user sectors in high demand seasons. In these market models, operational constraints of storage facilities are included. The storage capacities of European countries are available in Egging and Gabriel (2006). The related cost terms of storage facilities are available in Lise and Hobbs (2008).

2.5. Stochastic Network Design

Tometzki and Engell (2011) present a two stage stochastic MIP model for South Gabon oil pipeline problem. The stage decomposition approach by using multi objective algorithm is used to optimize the expected costs. Their purpose is to emphasize the optimization of both outcome and risk. Therefore, an hybrid multi objective algorithm is used to emphasize both objectives. In the first stage, the pipeline

layout is determined with uncertain oil production rates. After that, pipeline diameters are determined in the second stage. The stage decomposition approach by using multi objective algorithm is used to optimize expected scenario costs.

Zhang et al. (2017) present a stochastic MILP for optimal design of oil pipeline in large slope topology. Stochastic flow rates are included to determine the network structure. The model determines the location of network elements, pipeline diameters and optimal operating plan. Discrete diameter values are used within the model.

Butler and Dyer (1999) provide scenario optimization model for natural gas network. The scenarios are generated based on demand forecasts. The main aim is to generate operation model for electric companies. Companies decisions to buy or store the natural gas is included in the model.

2.6. Benders Decomposition

The Benders decomposition method has been used frequently for solving network design problems. The main idea behind the Benders decomposition is to decompose the problem into a master problem with complicating (integer) variables and a subproblem with continuous (easy) variables that are iteratively solved. In this section, a formal statement of Benders decomposition method for a mixed integer problem (MIP) is presented.

Consider the following problem:

$$\begin{aligned} &\text{Min } cx + dy \\ &Ax + By \geq b \\ &Dy \geq e \\ &x \geq 0 \\ &y \geq 0 \text{ and integer} \end{aligned} \tag{2.14}$$

where;

x	continuous variables
y	integer variables
A, B, D	coefficient matrices

b, e	right hand side vectors
c, d	row cost vectors

The problem (P) can be expressed equivalently as:

(P)

$$min_{\bar{y}\in Y}\{d\bar{y} + min_{x\geq 0}\{cx : Ax \geq b - B\bar{y}\}\} \qquad (2.15)$$

where $Y = \{y : Dy \geq e, y \geq 0$ and integer$\}$

Note that the inner minimization problem in (2.15) is a linear program, LP. If this LP is unbounded for some $y \in Y$ then P is unbounded. When the inner minimization is bounded, its dual can be written by the associating dual variables w. The dual of the inner minimization problem in (2.15) is called the subproblem (SP):

(SP)

$$\begin{aligned} \max\ & w\,(b - B\bar{y}) \\ & wA \leq c \\ & w \geq 0 \end{aligned} \qquad (2.16)$$

Then the problem P becomes:

$$min_{\bar{y}\in Y}\{d\bar{y} + max_{w\geq 0}\{w\,(b - B\bar{y}) : wA \leq c\}\} \qquad (2.17)$$

Note that the feasible space of (SP), $W = \{w : w \geq 0; wA \leq c\}$, is independent of the values of integer y variables.

SP is either bounded or unbounded. Infeasibility of SP implies the unboundedness of P. Assuming that problem P is not unbounded, it can be concluded that W is not empty and W has extreme rays ($r_1, r_2, ..., r_R$) and extreme points ($w_1, w_2, ..., w_Q$) where R and Q are the numbers of extreme rays and extreme points of W, respectively.

If SP is unbounded, then there is a direction r_r such that $r_r\,(b - B\bar{y}) > 0$ and r_r must be avoided. This condition is satisfied with:

$$r_r\,(b - B\bar{y}) \leq 0 \qquad r = 1, \dots, R \qquad (2.18)$$

These constraints are called "feasibility cuts".

If (SP) is bounded, then the solution is one of the extreme points w_q (q = 1,...,Q). Since the maximum value for (SP)'s over $y \in Y$ is to be found, any solution for (SP)

shall be less than or equal to objective function of SP. With a new continuous variable η, denoting the objective value of SP in (2.17), the following conditions should be met.

$$\eta \geq w_q(b - B\bar{y}) \qquad q = 1, \dots, Q \qquad (2.19)$$

These constraints are called "optimality cuts".

Rewriting (2.14) with additions of feasibility and optimality cuts, we have the Benders Decomposition's master problem (MP).

$$\begin{aligned} & Min\ dy + \eta \\ \eta \geq\ & w_q(b - B\bar{y}) && q = 1, \dots, Q \\ r_r(b - B\bar{y}) \leq\ & 0 && r = 1, \dots, R \qquad (2.19) \\ & y \in Y \\ & \eta \geq 0 \end{aligned}$$

Benders decomposition algorithm involves iterative solution of (MP) and (SP) to solve (P) by generating necessary optimality and feasibility cuts during the iterations. So, the relaxed (MP) is solved and an integer solution (y*, η*) is generated. (SP) is solved with this solution. If (SP) has an optimal solution and its objective function value is equal to η* then the algorithm stops. Otherwise, if the solution is bounded (unbounded), associated optimality (feasibility) cut is added to the relaxed (MP) and the relaxed (MP) is solved again. The solution of the relaxed (MP is a lower bound for (P). The solution of (SP) gives an upper bound for (P). The procedure stops at optimality when upper bound is equal to the lower bound. One can refer to Benders (1962) for more detailed information on Benders decomposition method. General procedure of Benders decomposition flow chart can be seen in Figure 2.1.

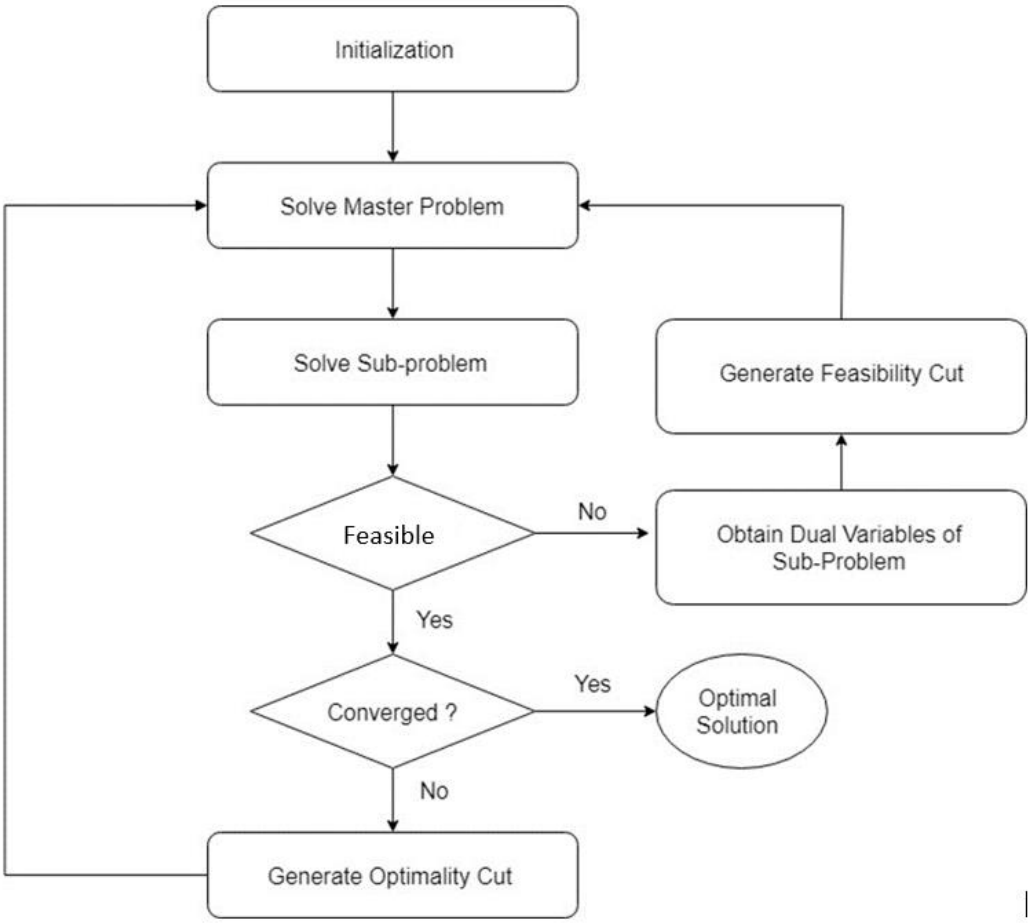

Figure 2.1. Benders Decomposition Flow Chart

2.7. Extension of Benders Decomposition

Since Benders decomposition method is introduced by Benders in 1962, several variants of the algorithm and improvement methods are proposed. Here some literature review is given on different variants and improvement methods. In the literature review, we mainly focus on improving Benders cuts thorough cut selection methods and improving master problem solution. Costa's literature survey (2005) is referred for more comprehensive survey on Benders decomposition methods for solving fixed-charge network design problems. Some of the studies related with these improvements are given in Table 2.1.

Miranda et al. (2011) formulates a multi commodity flow formulation for the local access network design problem with congestion cost. Mixed integer mathematical model generates minimal cost tree shape network design that connects switching center to demand node. The model is basically multi commodity flow problem and solved by Benders decomposition. The constraints which aim to generate tree structure from root node is added to relaxed master problem (RMP). Previous constraints do not prevent the formation of cycles so that cycle prevention constraints are also added to RMP.

D'Addio et al. (2017) analyzes long term electricity distribution network expansion planning problem. The convex outer approximation and Benders decomposition are applied to model by using lazy constraints. Lazy constraints turn the complex problems into simple format. This technique allows certain constraints to be out of model until it is needed. The convex nonlinear loss cost function is approximated with outer approximation technique. The hyperplane cuts are appealed to provide bounds of the loss cost of each arc.

Lee et al. (2013) apply Benders decomposition into network design problems with stochastic node demands. The model constructs facilities on the edges of network so that the capacities of facilities define the flows on the edges. The simultaneous benders cut generation procedure is applied to decrease the number of iterations. The procedure tests many solutions simultaneously to generate the cuts that violates all the points. In addition, they also apply cut selection procedure for benders cut based on minimal infeasible subsystem of a linear programming problem. Classical Benders cut which corresponds to a small cardinality are selected by minimal infeasible subsystems (MIS). Restriction on summation of dual variables provides small cardinality.

Wheatley et al. (2015) apply logic-based benders decomposition to integrated inventory location problem. Mathematical model specifies the location decisions in the master problem so sub-problem determines inventory decisions. It minimizes the fixed cost of facility installation, transportation and inventory cost. The nonlinearity in the sub-problem structure makes it impossible to use linear duality theory so that

logic based Benders cuts are generated for the fill rate parameter. It provides to isolate inventory decisions from location and assignment decisions. It is stated that obtaining the solution becomes difficult as more logic cuts are added to problem. Service centers cuts are appealed after logic based cuts to generate another set service location

Mahey et al. (2001) solves capacitated flow problem in telecommunication networks. Master problem initialize the capacity of link so that sub problem becomes flow assignment problem. Valid inequalities are added to the model to decrease the number of iterations. The valid inequalities simply check the flow and capacity relationship. The model also includes spanning tree and connectivity. Connectivity cuts check whether there exists at least one path between each origin and destination pairs.

Binato et al. (2001) uses Gomory cuts for power transmission network design. Former sub-problem defines Gomory cuts for solution of the next sub-problem. The Benders cuts will not produce successful results due to large values of disjunctive parameter. Therefore, they update the Benders cut coefficients according to the values of parameter.

Randazzo et al. (2001) models multi commodity flow of local access network design. The model minimizes initialization cost of arcs and variable cost of flows. Structural constraints which yields arborescence structure is added to decrease the number of iteration in Benders. The structural constraints include cycle prevention.

Gendron et al. (2014) considers branch and Benders cut method for design of wireless local area network to decrease the power consumption. The model determines the user assignments to access points and power level of each point which depends on the number of users. The sub-problem structure is not a LP so that LP based duality cuts cannot be applied. Canonical cuts are applied to the model.

Ergün (2013) solves the telecommunication network design problem to satisfy demand and flow with respect to constraints of topology, capacity and technology constraints in her dissertation. The dissertation includes a wider literature review related with the cut selection methods. They apply the cut selection procedure to generate the most violated Benders cut through each iteration. In addition, violated bipartition cuts are added to master problem to obtain integer feasible solution.

Çakır (2009) models the multi commodity, multi node supply chain problem so that it is not required to yield solutions which is in the shape of tree structure. Mixed integer problem is formulated and solved with Benders decomposition. Capacitated supply chain problem includes fixed and variable cost related with routing over links. It is stated that the upper and lower bounds of large problems cannot converge each other in a reasonable number of iterations. Ɛ- optimality and Pareto-optimal cut generation can be applied to reach acceptable results. Oliveira et al. (1995) solves MINLP of transmission network expansion problem. Logical constraints are included to incorporate relationship between decisions variables. The summary of the above studies can be seen in Table 2.1.

Table 2.1. *Summary of Studies Related with Benders Decomposition*

Reference	Type of Problem	Additional Constraints	Additional Method
Miranda et al. (2011)	Local access network design	Cycle prevention, absorbance structure constraints	
D'Addio et al. (2017)	Long term electricity distribution network expansion planning problem	Hyperplane constraints to estimate lower bound for nonlinear function	Convex outer approximation using lazy constraints
Lee et al. (2013)	Capacitated stochastic network design problem (MIP)	Cut set inequalities to verify capacity and flow relation in arcs	Simultaneous cut generation, minimal infeasible subsystems
Randazzo et al. (2001)	Extension of local access network design	Cycle prevention and absorbance structure constraints	Randazzo et al. (2001)
Gendron et al. (2014)	Design of wireless network	Canonical cuts due to nonlinear relaxed problem	Branch and cut algorithm to solve master problem

Reference	Type of Problem	Additional Constraints	Additional Method
Binato et al. (2001)	Power transmission network expansion planning problem (MIP)	Gomory cuts	Updating the previously evaluated Benders cuts, three benders cut generated in one iteration
Wheatley et al. (2015)	Inventory location problem	Logic based Benders cuts for nonlinear equations of relaxed problem	Restrict and decompose scheme to restrict solution space
Mahey et al. (2001)	Capacitated network flow problem	Absorbance and connectivity constraints to check path between each origin and destination pairs, capacity check constraints based on flow on arc	Proximal decomposition method
Ergün (2013)	Telecommunication network design (MIP)	Bipartition cuts to master problem	Cut selection and branch and cut procedure, heuristic solution to master and sub-problem
Oliveira et al. (1995)	Transmission network expansion (MINLP)	Logical cuts to incorporate relation between decision variables of sub-problem	Branch and bound algorithm to solve sub-problem

2.8. Stochastic Programming

Stochastic programming is a kind of mathematical programming in which there exist uncertainties related with some parameters. Dantzig (1955) is the first one to combine linear programing with uncertainty. Uncertain parameters are defined by the known probability distributions so that the solution that is feasible to all cases will be generated. Stochastic programing generates the expected value of the objective function.

2.8.1. Two Stage Stochastic Programing

Two-stage stochastic programming is the basic version of stochastic modelling. It consists of two stages. In the first stage, there exist an uncertainty so that first stage decisions affect the second stage decisions. The uncertainties are solved in the second stage that the decisions are made for the observed scenarios. Birge and Louveaux (2011) provide the mathematical model of two stage stochastic program as follows:

$$\begin{aligned} \min c^T x + E_\xi[q(w)^T q(w)] \\ Ax = b \\ T(w)x + Wy(w) = h(w) \\ x \geq 0\,, y(w) \geq 0 \end{aligned} \tag{2.20}$$

The first stage decisions are determined by the x vector. The second stage decisions are determined by the y vector. $\xi(w)$ presents the random parameters which are determined by scenario. First and second stage decisions are connected with random events. The expected cost of the second stage is included in the model. The expected value of the second stage determined by cost coefficient of second stage q and second stage decisions y. There exists stochastic part for each scenario w. $T(w)$ is called as technology matrix for each scenario. The stochastic right hand side vector of second stage is presented by $h(w)$. The $c^T x$ term in objective function generates the deterministic part of the objective function. $E_\xi[q(w)^T q(w)]$ is the expectation over all realized random parameters w. Birge and Louveaux (2011) states deterministic equivalent program is as follows:

$$\begin{aligned} &\min c^T x + \mathfrak{Q}(\mathrm{x}) \\ &Ax = b \\ &x \geq 0 \end{aligned} \tag{2.21}$$

Where $\mathfrak{Q}(\mathrm{x}) = E_{\xi}[Q(x, \xi(w))]$

$Q(x, \xi(w)) = min_y\{q(w)^T y | W(w) = h(w) - T(w)x, y \geq 0\}$ is called as the second stage value function. Extensive form of stochastic model is obtained by generating $Q\big(x, \xi(w)\big)$ for all scenarios. Birge and Louveaux (2011) states the discretized version of deterministic equivalent program as follows:

$$\begin{aligned} &\min c^T x + \sum_{k=1}^{K} p_k q_k^T y_k \\ &Ax = b \\ &T_k x + W y_k = h_k && k = 1, \dots, K \\ &x \geq 0, y_k \geq 0 && k = 1, \dots, K \end{aligned} \tag{2.22}$$

Scenario index is represented by *k*.

2.8.2. L-Shaped Method

The L-shaped method is used to approximate the second stage objective function. The method divides the problem into two problems which are the main problem and the sub problem. The main problem determines the first stage decisions. The sub-problem is solved for given first stage solutions. Main structure of the method is similar to Benders Decomposition. The master problem is solved to optimality. After that, sub problem is solved for given first stage decisions. In each iteration, feasibility or optimality cuts are added to the master problem. The L-shaped method is the stochastic version of Benders decomposition.

CHAPTER 3

NETWORK DESING FOR THE NATURAL GAS PIPELINE SYSTEM

In this chapter, two models are developed to create a network design for the natural gas pipeline system. The pipeline network includes natural gas supply (or entrance) nodes, demand nodes and storage nodes. A possible pipeline network configuration can be seen in Figure 3.1. In this figure, pipeline network consists of entrance, demand and storage nodes. Imported gas enter the system from the entrance nodes (ports). Demand nodes receive natural gas from upstream nodes to meet its own demand. Remaining gas is transferred to downstream elements of the pipeline network. Natural gas can directly bypass the storage node without being stored so that it is directly transferred to the next node in the network structure. Or, it can be stored in the storage nodes for certain time periods so that it will be used in the upcoming time periods. Natural gas can be stored in gaseous and liquefied form.

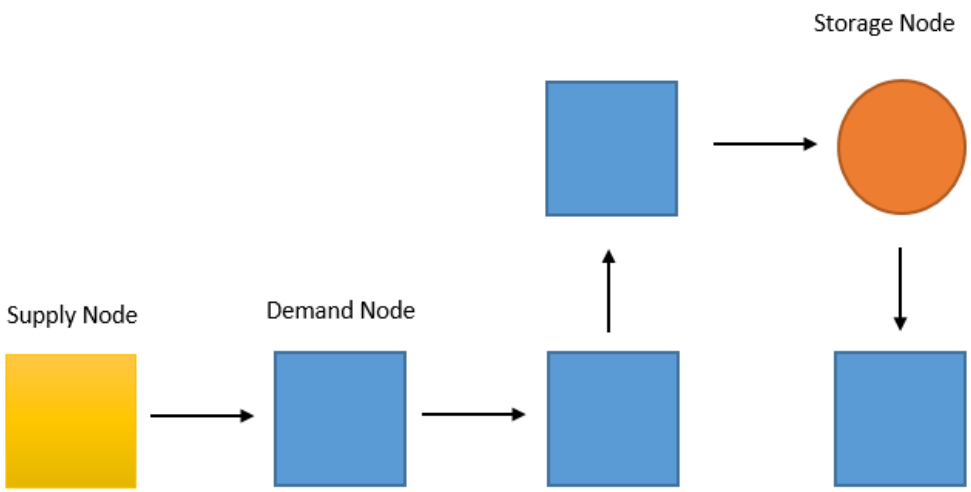

Figure 3.1. Pipeline Network Configuration

The pipeline network design problem analyzed in this study determines the network structure of the pipeline system, and whether/where to have storage units. This problem is named as Pipeline and Storage Network Design Problem, PSNDP.

Natural gas enters the system from the entrance nodes. There exists an upper bound for supply for each entrance node. The imported gas is distributed to demand nodes using the pipeline system. The demand of all the demand nodes should be satisfied for all time periods.

There exist a set P containing the entrance nodes through which gas is received, a set M consisted of cities or demand nodes, and a set K consisted of possible storage sites for natural gas. The pipeline network is modeled as an directed network G = (N,A) with node set N and arc set A. The node set $N = \{\{0\} \cup P \cup M \cup K\}$, where node 0 is the root node. The arc set A includes possible connections between the nodes considering the proximity between the nodes and geographic conditions. The problem is studied for a time horizon of T periods.

In defining the pipeline system, the following decisions are made:

a) Which arcs of the network will be used to establish a pipeline
b) How much flow will be sent through these arcs in different time periods
c) Based on these flows, what should be the diameters of each pipe
d) How many and which of the storage units should be opened
e) What should be the capacity of these storage units.

The problem has the characteristics of the one-terminal Telpak problem with fixed and variable costs associated with arcs. The variable costs of the pipes on the arcs depend on the diameter of the pipes, which are affected by the flow on the pipes. The infrastructure costs are considered to be the monthly equivalents calculated over the life cycle of the system.

Two versions of PSNDP are developed in the next sections. First a deterministic version of PSNDP, named as PSNDP-D is developed for the case where the demands

of the demand nodes are considered to be known with certainty for all time periods. Then a stochastic version of the problem, named as PSNDP-S, is developed for the case where the demands of the demand nodes are considered to be uncertain.

Detailed pipeline flow equations are available in Appendix-D.

3.1. Deterministic Mathematical Model, PSNDP-D

3.1.1. Parameters

γ_{ij}: the distance of the arc (i,j) (km)

c_s: monthly equivalent fixed cost of opening a storage facility ($/month)

c_o : monthly equivalent fixed cost of establishing a pipe ($/km-month)

c_p : monthly equivalent cost of a pipe per diameter, per km ($/month-inch-km)

c_{bt} : purchasing cost of gas at month *t* ($/MMscfd)

c_{ht} : holding cost of gas at storage in month *t* ($/MMscfd/month)

d_{it} : demand of city *i* in month *t* (MMscfd)

s_{pt} : supply at entrance node *p* in month *t* (MMscfd)

β : conversion parameter of flow on an arc to the diameter of the arc

$\overline{C_k}$: capacity of storage node *k*

u : flow limit in each pipeline segment

b_i : spanning tree construction parameter

B : big M constant for pipe construction

3.1.2. Decision Variables

y_{ij}: $\begin{cases} 1, \text{if a pipe is established on arc } (i,j) \\ 0, \text{otherwise} \end{cases}$

x_{ij}: flow of commodity on arc (i,j) while constructing the pipeline network

o_k: $\begin{cases} 1, \text{if storage facility } k \text{ is opened} \\ 0, \text{otherwise} \end{cases}$

f_{ijt}: flow of gas on arc (i,j) in month t

g_{kt}: Amount stored at node k in month t

Δ_{ij}: Diameter of pipe for arc (i,j)

$$Minimize \sum_{t\in T}\sum_{(i,j)\in A} c_o\gamma_{ij}y_{ij} + \sum_{t\in T}\sum_{(i,j)\in A} c_p\gamma_{ij}\Delta_{ij} + \sum_{t\in T}\sum_{k\in K} c_s o_k + \sum_{t\in T}\sum_{k\in K} c_{ht}g_{kt} + \sum_{t\in T}\sum_{p\in P}\sum_{(p,j)\in A} c_{bt}f_{pjt} \quad (3.2)$$

$$\sum_{(i,j)\in A} x_{ij} - \sum_{(k,i)\in A} x_{ki} = b_i \qquad \forall i \in N\backslash\{0\} \quad (3.3)$$

$$\sum_{(0,j)\in A} x_{0j} \le b_0 \quad (3.4)$$

$$x_{ij} \le By_{ij} \qquad \forall (i,j) \in A \quad (3.5)$$

$$x_{ij} \le By_{ji} \qquad \forall (i,j) \in A \quad (3.6)$$

$$\sum_{(p,j)\in A} f_{pjt} \le s_{pt} \qquad \forall p \in P, \ \forall t \in T \quad (3.7)$$

$$\sum_{(j,i)\in A} f_{jit} - \sum_{(i,j)\in A} f_{ijt} = d_{it} \qquad \forall i \in M, \ \forall t \in T \quad (3.8)$$

$$g_{kt-1} + \sum_{(j,k)\in A} f_{jkt} = g_{kt} + \sum_{(k,j)\in A} f_{kjt} \qquad \forall k \in K, \ \forall t \in T \quad (3.9)$$

$$g_{kt} \leq \overline{C_k'}\, o_k \qquad \forall k \in K,\ \forall t \in T \tag{3.10}$$

$$f_{ijt} \leq u y_{ij} \qquad \forall (i,j) \in A,\ \forall t \in T \tag{3.11}$$

$$\beta f_{ijt} \leq \Delta_{ij} \qquad \forall (i,j) \in A,\ \forall t \in T \tag{3.12}$$

$$x_{ij} \geq 0 \qquad \forall (i,j) \in A \tag{3.13}$$

$$y_{ij} \in \{0,1\} \qquad \forall (i,j) \in A \tag{3.14}$$

$$o_k \in \{0,1\} \qquad \forall k \in K \tag{3.15}$$

$$\Delta_{ij} \geq 0 \qquad \forall (i,j) \in A \tag{3.16}$$

$$g_{kt} \geq 0 \qquad \forall k \in K, t \in T \tag{3.17}$$

$$f_{ijt} \geq 0 \qquad \forall (i,j) \in A,\quad t \in T \tag{3.18}$$

The objective function minimizes five terms. These are the fixed costs of establishing pipelines, fixed costs of establishing pipes of given diameters, fixed costs of opening a storage site, natural gas holding cost, and purchasing cost of natural gas. Spanning tree structure is used to ensure connected network structure for pipeline system. Equations (3.3) and Equations (3.4) are used to generate spanning tree structure in the networks. Total number of arcs that is going out from the root node is restricted by Equations (3.4). Equations (3.5) and Equations (3.6) defines the arcs on the spanning tree. Equations (3.7) are the supply constraints. Equations (3.8) are the demand constraints. Equations (3.9) provide flow balance for storage nodes. Equations (3.10) is related with opening decisions of storage areas. Equations (3.11) state that only opened pipes can provide flow at all time periods. Equations (3.12) determine the biggest flow of gas in routed pipe throughout all time periods. The biggest flow value is used to determine the pipe diameter. Equations (3.13), (3.16), (3.17) and (3.18) ensure that variables will be positive. Equations (3.14) and (3.15) ensure that binary decisions are made.

3.2. Two Stage Stochastic Mathematical Model PSNDP-S

Stochastic version of mathematical model is generated with scenario indexes. There exist a set ϕ (w ∈ ϕ) representing scenarios for the problem. The mathematical model parameters and variables are stated as below. In this mathematical model, there exist uncertainties in demands of nodes. The first stage decisions are made with uncertainties of these parameters. PSNDP-S is similar to PSNDP-D, with the only difference of scenario indexes on demand values of demand nodes and on the decision variables. Therefore, related explanations of constraints are available at section 3.1.3.

In our mathematical model, decisions are to be made now corresponds to first stage decisions. After realization of uncertainties, second stage decisions are generated. Pipeline arcs between nodes and storage unit opening decisions belong to the first stage. Network flow decisions belongs to second stage and are determined after realization of demand scenarios. The main aim of two stage stochastic model is to generate a solution that operates well under scenarios.

3.2.1. Mathematical Model Parameters

γ_{ij}: the distance of the arc (i, j) (km)

$prob_w$: probability of scenario *w*

c_s: monthly equivalent fixed cost of opening a storage facility ($/month)

c_o : monthly equivalent fixed cost of establishing a pipe ($/km-month)

c_p : monthly equivalent cost of a pipe per diameter, per km ($/month-inch-km)

c_{bt} : purchasing cost of gas at month *t* ($/MMscfd)

c_{ht} : holding cost of gas at storage in month *t* ($/MMscfd/month)

β : conversion parameter of flow on an arc to the diameter of the arc

${d_{it}}^w$: demand of node *i* in month *t* (MMscfd) in scenario *w*

s_{pt}: supply of port p in month t (MMscfd)

$\overline{C_k}$: capacity of storage point k

u : flow limit in each pipeline segment

b_i : spanning tree construction parameter

B : big M constant for pipe construction

3.2.2. Mathematical Model Decision Variables

First stage decision variables are as below.

$$y_{ij}: \begin{cases} 1, \text{if a pipe is established on arc } (i,j) \\ 0, \text{otherwise} \end{cases}$$

x_{ij}: flow of commodity on arc (i,j) while constructing the pipeline network

$$o_k: \begin{cases} 1, \textit{if} \text{ storage facility k is opened} \\ 0, \text{otherwise} \end{cases}$$

Second stage decision variables are stated as below.

${f_{ijt}}^w$: flow of gas on arc (i, j)in month t in scenario w

${g_{kt}}^w$: Amount stored at node k in month t in scenario w

Δ_{ij}: Diameter of pipe for arc (i,j)

3.2.3. First Stage Mathematical Model

$$Minimize \sum_{t \in T} \sum_{(i,j) \in A} c_o \gamma_{ij} y_{ij} + \sum_{t \in T} \sum_{k \in K} c_s o_k \tag{3.19}$$

$$\sum_{(i,j) \in A} x_{ij} - \sum_{(k,i) \in A} x_{ki} = b_i \qquad \forall i \in N \backslash \{0\} \tag{3.20}$$

$$\sum_{(0,j) \in A} x_{0j} \leq b_0 \tag{3.21}$$

$$x_{ij} \leq By_{ij} \qquad \forall(i,j) \in A \qquad (3.22)$$

$$x_{ij} \leq By_{ji} \qquad \forall(i,j) \in A \qquad (3.23)$$

$$x_{ij} \geq 0 \qquad \forall(i,j) \in A \qquad (3.24)$$

$$y_{ij} \in \{0,1\} \qquad \forall(i,j) \in A \qquad (3.25)$$

$$o_k \in \{0,1\} \qquad \forall k \in K \qquad (3.26)$$

3.2.4. Second Stage Mathematical Model

In the second stage, the arcs with established pipes are stored in the set A^w, and the selected storage facilities are stored in the set K^w.

$$Minimize \sum_{t\in T}\sum_{(i,j)\in A} c_p\gamma_{ij}\Delta_{ij} + \sum_{w\in\phi}\sum_{t\in T}\sum_{k\in K} prob_w c_{ht} {g_{kt}}^w + \sum_{w\in\phi}\sum_{t\in T}\sum_{p\in P}\sum_{(p,j)\in A} prob_w c_{bt} {f_{pjt}}^w \qquad (3.27)$$

$$\sum_{(p,j)\in A} {f_{pjt}}^w \leq s_{pt} \qquad \forall p \in P, \forall w \in \phi \quad \forall t \in T \qquad (3.28)$$

$$\sum_{(j,i)\in A} {f_{jit}}^w - \sum_{(i,j)\in A} {f_{ijt}}^w = {d_{it}}^w \qquad \forall i \in M, \forall w \in \phi \quad \forall t \in T \qquad (3.29)$$

$${g_{kt-1}}^w + \sum_{(j,k)\in A} {f_{jkt}}^w = {g_{kt}}^w + \sum_{(k,j)\in A} {f_{kjt}}^w \qquad \forall k \in K^w \quad \forall w \in \phi \quad \forall t \in T \qquad (3.30)$$

$${g_{kt}}^w \leq \overline{C_k'} \qquad \forall k \in K^s, \forall w \in \phi \quad \forall t \in T \qquad (3.31)$$

$$f_{ijt}{}^{w} \leq u \qquad \forall (i,j) \in A^{w}, \forall w \in \phi, \forall t \in T \qquad (3.32)$$

$$\beta\, f_{ijt}{}^{w} \leq \Delta_{ij} \qquad \forall (i,j) \in A^{w}, \forall w \in \phi, \forall t \in T \qquad (3.33)$$

$$\Delta_{ij} \geq 0 \qquad \forall (i,j) \in A^{w} \qquad (3.34)$$

$$g_{kt}{}^{w} \geq 0 \qquad \forall w \in K^{w}, t \in T \qquad (3.35)$$

$$f_{ijt}{}^{w} \geq 0 \qquad \forall (i,j) \in A^{w}, t \in T \qquad (3.36)$$

3.3. Solution Procedures for PSNDP-D

In Section 2.6, we have stated the basic structures of the Benders decomposition method. As was stated there, Benders decomposition method has been used very often for solving network flow problems. Thus, for solving PSNDP-D, we consider the application of the method, and its extensions.

3.3.1. Benders Decomposition Application to PSNDP-D

After solving the RMP problem in iteration *I*, the arcs with established pipes are stored in the set A^{I}, and the opened storage nodes are stored in the set K^{I}. The sub-problem is stated as below.

(SP)

$$Minimize \sum_{(i,j)\in A^{I}} c_p \gamma_{ij} \Delta_{ij} + \sum_{t\in T} \sum_{p\in P} \sum_{(p,j)\in A^{I}} c_{bt} f_{pjt} + \sum_{t\in T} \sum_{k\in K^{I}} c_{ht} g_{kt} \qquad (3.37)$$

$$\sum_{(p,j)\in A^{I}} f_{pjt} \leq s_{pt} \qquad \forall p \in P,\ \forall t \in T \qquad (3.38)$$

$$\sum_{(j,i)\in A^{I}} f_{jit} - \sum_{(i,j)\in A^{I}} f_{ijt} = d_{it} \qquad \forall i \in M,\ \forall t \in T \qquad (3.39)$$

$$g_{kt-1} + \sum_{(j,k)\in A^I} f_{jkt} - g_{kt} - \sum_{(i,j)\in A^I} f_{kjt} = 0 \qquad \forall k \in K^I \quad \forall t \in T \tag{3.40}$$

$$g_{kt} \leq \overline{C_k} \qquad \forall k \in K^I \quad \forall t \in T \tag{3.41}$$

$$f_{ijt} \leq u \qquad \forall (i,j) \in A^I \quad \forall t \in T \tag{3.42}$$

$$\beta f_{ijt} - \Delta_{ij} \leq 0 \qquad \forall (i,j) \in A^I \quad \forall t \in T \tag{3.43}$$

$$f_{ijt} \geq 0 \qquad \forall (i,j) \in A^I, \quad t \in T \tag{3.44}$$

$$g_{kt} \geq 0 \qquad \forall k \in K^I, t \in T \tag{3.45}$$

$$\Delta_{ij} \geq 0 \qquad \forall (i,j) \in A^I \tag{3.46}$$

For the Ith iteration, the dual variables, D^I_{1pt}, D^I_{2it}, D^I_{3kt}, D^I_{4kt}, D^I_{5ijt}, D^I_{6ijt} , (for $\forall p \in P, \forall i \in M$, $k \in K^I$, $\forall (i,j) \in A^I$, and $, t \in T$) are defined for constraints (3.38)-(3.43), respectively.

Then the optimality cut is determined as follows:

$$\eta \geq \sum_{p\in P}\sum_{t\in T} D^I_{1pt}\, s_{pt} + \sum_{i\in M}\sum_{t\in T} D^I_{2it}\, d_{it} + \sum_{k\in K^I}\sum_{t\in T} D^I_{4kt}\, \bar{C}_k + \sum_{(i,j)\in A^I}\sum_{t\in T} D^I_{5ijt}\, u \tag{3.47}$$

When the (SP) has no feasible solution, an extreme ray should be obtained from the (SP). Marin and Jaramillo (2009) points out that this extreme ray can be obtained in GAMS environment by solving a modified dual subproblem, (MDSP) in which a dummy objective function is defined, and the objective function of the dual problem is set to one, with constraints of the dual problem also modified. The modified dual

subproblem (MDSP) for iteration I is provided below with dual variables q_{1pt}^{I}, q_{2it}^{I}, q_{3kt}^{I}, q_{4kt}^{I}, q_{5ijt}^{I}, q_{6ijt}^{I} , (for $\forall p \in P, \forall i \in M$, $k \in K^I$, $\forall (i,j) \in A^I$, and , $t \in T$).

(MDSP)

$$Maximize\ \Delta \tag{3.48}$$

$$-\sum_{p\in P}\sum_{t\in T} q_{1pt}^{I}\, s_{pt} - \sum_{i\in M}\sum_{t\in T} q_{2it}^{I}\, d_{it} - \sum_{k\in K^I}\sum_{t\in T} q_{4kt}^{I}\, \bar{C}_k - \sum_{(i,j)\in A^I}\sum_{t\in T} q_{5ijt}^{I}\, u = 1 \tag{3.49}$$

$$\Delta = 0 \tag{3.50}$$

$$\sum_{t=1}^{m} q_{6ijt}^{I} \le 0 \qquad \forall (i,j) \in A^I \tag{3.51}$$

$$q_{3kt}^{I} - q_{4kt}^{I} - q_{3kt+1}^{I} \le 0 \qquad \forall k \in K^I \tag{3.52}$$

$$-q_{1pt}^{I} - q_{2it}^{I} - q_{5ijt}^{I} - q_{6ijt}^{I} \le 0 \qquad \forall (i,j) \in A^I, \forall t \in T \tag{3.53}$$

$$-q_{2jt}^{I} + q_{2it}^{I} - q_{5ijt}^{I} - q_{6ijt}^{I} - q_{3kt}^{I} \le 0 \qquad \forall (i,j) \in A^I, \forall t \in T \tag{3.54}$$

$$-q_{2jt}^{I} - q_{5kjt}^{I} - q_{6kjt}^{I} + q_{3kt}^{I} \le 0 \qquad \forall (k,j) \in A^I, \forall k \in K^I \quad \forall t \in T \tag{3.55}$$

$$q_{1pt}^{I} \ge 0,\ q_{2it}^{I}\ urs,\ q_{3kt}^{I}\ urs,\ q_{4kt}^{I} \ge 0,\ q_{5ijt}^{I} \ge 0, q_{6ijt}^{I} \ge 0 \tag{3.56}$$

Then the feasibility cut for iteration I is given by,

$$\sum_{p\in P}\sum_{t\in T} q_{1pt}^{I}\, s_{pt} + \sum_{i\in M}\sum_{t\in T} q_{2it}^{I}\, d_{it} + \sum_{k\in K^I}\sum_{t\in T} q_{4kt}^{I}\, \bar{C}_k + \sum_{(i,j)\in A^I}\sum_{t\in T} q_{5ijt}^{I}\, u \le 0 \tag{3.57}$$

The RMP in iteration I is stated as below. Equation (3.59) is the optimality cut of problem. Equation (3.60) are the feasibility cut of problem. Equation (3.59) are used to generate the spanning tree.

$$Minimize \sum_{(i,j)\in A} c_o \gamma_{ij} y_{ij} + \sum_{(i,j)\in A} c_p \gamma_{ij} \Delta_{ij} + \sum_{k\in K} c_w o_k + \eta \tag{3.58}$$

Subject to (3.3)- (3.6) and (3.13)-(3.15)

$$\eta \geq \sum_{p\in P}\sum_{t\in T} w_{1pt}^{\vartheta} s_{pt} + \sum_{i\in M}\sum_{t\in T} w_{2it}^{\vartheta} d_{it} + \sum_{k\in K^I}\sum_{t\in T} w_{4kt}^{\vartheta} \bar{C}_k + \sum_{(i,j)\in A^I}\sum_{t\in T} w_{5ijt}^{\vartheta} u \qquad \vartheta = 1,\ldots,I \tag{3.59}$$

$$\sum_{p\in P}\sum_{t\in T} q_{1pt}^{\vartheta} s_{pt} + \sum_{i\in M}\sum_{t\in T} q_{2it}^{\vartheta} d_{it} + \sum_{k\in K^I}\sum_{t\in T} q_{4kt}^{\vartheta} \bar{C}_k + \sum_{(i,j)\in A^I}\sum_{t\in T} q_{5ijt}^{\vartheta} u \leq 0 \qquad \vartheta = 1,\ldots,I \tag{3.60}$$

$$\eta \geq 0 \tag{3.61}$$

3.3.2. Benders Decomposition with Extreme Rays (PSNDP-B1)

The first method used as the solution procedure is the one explained above as the classic Benders decomposition method. The feasibility cuts are generated from the modified subproblem. The optimality cuts are generated by using the dual variables of the subproblem. The general procedure of Benders Decomposition with Extreme Rays can be seen in Figure 3.2.

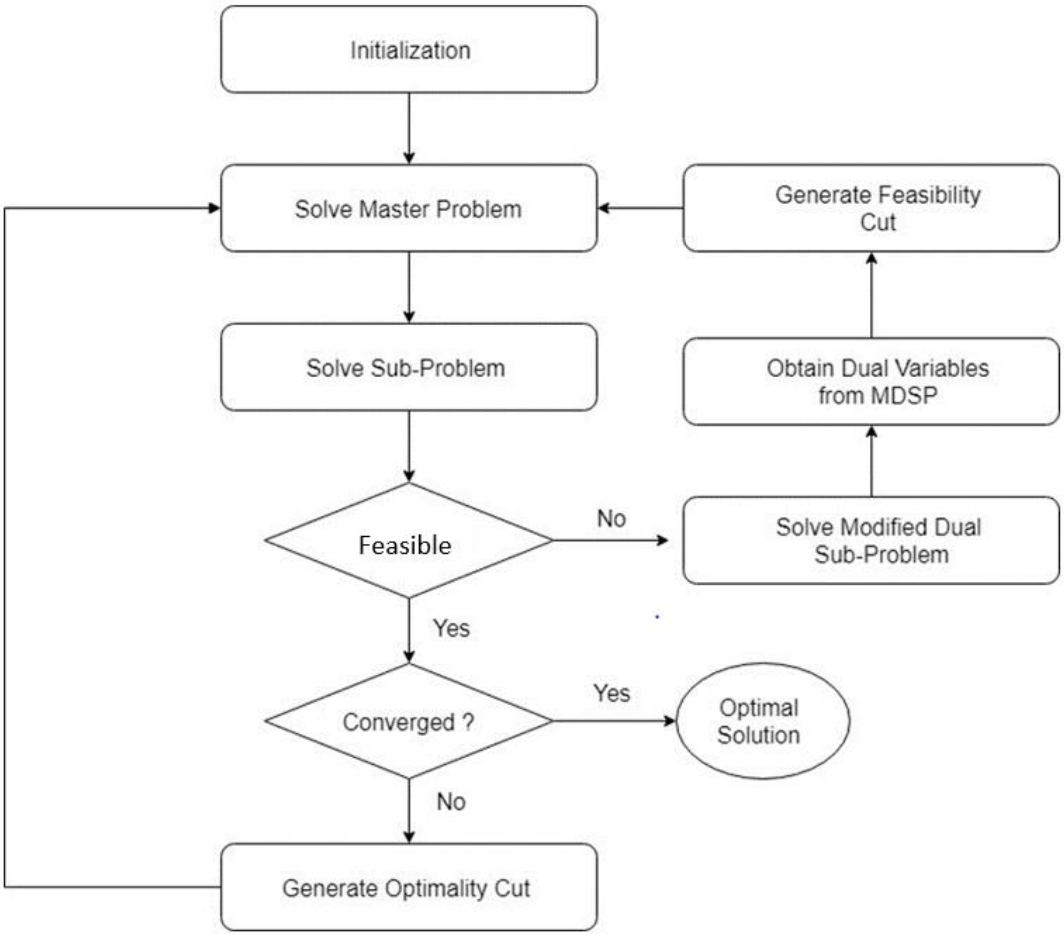

Figure 3.2. Flowchart for Benders Decomposition with Extreme Rays Algorithm

3.3.3. Benders Decomposition with Extreme Points (PSNDP-B2)

Wu et. al. (2003) explains Benders decomposition by using extreme points. The feasibility and optimality cuts are generated directly from the dual variables corresponding to the sub-problem. The mathematical model of the sub problem can be seen below.

(SP)

$$Minimize \sum_{(i,j)\in A^I} c_p \gamma_{ij} \Delta_{ij} + \sum_{t\in T} \sum_{k\in K^I} c_{ht} g_{kt} + \sum_{t\in T} \sum_{p\in P} \sum_{(p,j)\in A^I} c_{bt} f_{pjt} \quad (3.62)$$

$D^I_{1pt} \geq 0$ $$\sum_{(p,j)\in A^I} f_{pjt} \leq s_{pt} \quad \forall p \in P, \forall t \in T \quad (3.63)$$

D^I_{2it} urs $$\sum_{(j,i)\in A^I} f_{jit} - \sum_{(i,j)\in A^I} f_{ijt} = d_{it} \quad \forall i \in M, \forall t \in T \quad (3.64)$$

D^I_{3kt}urs $$g_{kt-1} + \sum_{(j,k)\in A^I} f_{jkt} - g_{kt} - \sum_{(i,j)\in A^I} f_{kjt} = 0 \quad \forall k \in K^I, \forall t \in T \quad (3.65)$$

$D^I_{4kt} \geq 0$ $$g_{kt} \leq \acute{\overline{C_k}} \quad \forall k \in K^I, \forall t \in T \quad (3.66)$$

$D^I_{5ijt} \geq 0$ $$f_{ijt} \leq u \quad \forall (i,j) \in A^I, \forall t \in T \quad (3.67)$$

$D^I_{6ijt} \geq 0$ $$\beta f_{ijt} - \Delta_{ij} \leq 0 \quad \forall (i,j) \in A^I, \forall t \in T \quad (3.68)$$

D^I_{1pt}, D^I_{2it}, D^I_{3kt}, D^I_{4kt}, D^I_{5ijt} and D^I_{6ijt} are the dual variables corresponding to the equations of the sub-problem respectively. The feasibility cut of the model is stated in (3.69).

$$\sum_{p\in P} \sum_{t\in T} D^I_{1pt}\, s_{pt} + \sum_{i\in M} \sum_{t\in T} D^I_{2it}\, d_{it} + \sum_{k\in K^I} \sum_{t\in T} D^I_{4kt}\, \bar{C}_k + \sum_{(i,j)\in A^I} \sum_{t\in T} D^I_{5ijt}\, u \leq 0 \quad (3.69)$$

Optimality cut of problem is stated as (3.70)

$$\eta \geq \sum_{p\in P}\sum_{t\in T} D^I_{1pt}\, s_{pt} + \sum_{i\in M}\sum_{t\in T} D^I_{2it}\, d_{it} + \sum_{k\in K^I}\sum_{t\in T} D^I_{4kt}\, \bar{C}_k + \sum_{(i,j)\in A^I}\sum_{t\in T} D^I_{5ijt}\, u \tag{3.70}$$

The general procedure of Benders Decomposition with Extreme Points can be seen in Figure 2.1

3.3.4. Benders Decomposition with Maximum Feasible Subsystem Cut Generation (PSNDP-B3)

Maximum feasible sub-system cut (MFS) generation procedure is proposed by Saharidis and Ierapetritou (2010). MFS searches for an alternative extreme point which will be produced by a problem which is constructed based on the infeasible (SP) with the minimum number of modifications in its infeasible solution space. The idea is to relax infeasible constraints with a minimum number of modifications. The extended subproblem (ESP) is relaxed as shown below.

$$\begin{aligned}
&Minimize \sum_{k\in K}\sum_{t\in T} \pi 1_{kt} + \sum_{(i,j)\in A^I}\sum_{t\in T} \pi 2_{ijt} + \sum_{i\in M}\sum_{t\in T} \pi 3_{it} && \\
&\sum_{(p,j)\in A} f_{pjt} \leq s_{pt} && \forall p \in P \\
& && \forall t \in T \\
&\sum_{(j,i)\in A} f_{jit} - \sum_{(i,j)\in A} f_{ijt} + d_{it}\, \pi 3_{it} \geq d_{it} && \forall i \in M \\
& && \forall t \in T \\
&g_{kt-1} + \sum_{(j,k)\in A} f_{jkt} = g_{kt} + \sum_{(k,j)\in A} f_{kjt} && \forall k \in K^I \\
& && \forall t \in T \\
&g_{kt} \leq \overline{C'_k} + \pi 1_{kt} * BigM && \forall k \in K^I \\
& && \forall t \in T \\
&f_{ijt} \leq u + \pi 2_{ijt} * BigM && \forall (i,j) \in A^I \\
& && \forall t \in T
\end{aligned} \tag{3.71}$$

$$\beta f_{ijt} - \Delta_{ij} \leq 0 \qquad \forall (i,j) \in A^I, \ \forall t \in T$$

$$\pi 1_{kt}, \pi 2_{ijt}, \pi 3_{it} \in \{0,1\} \qquad \forall k \in K^I, \ \forall (i,j) \in A^I$$

The system of inequalities (3.71) is solved with additional binary variables. The system (3.72) is solved with the correct values of binary variables in order to generate optimality cut.

$$\begin{aligned}
&Minimize \sum_{(i,j)\in A^I} c_p \gamma_{ij} \Delta_{ij} + \sum_{t\in T} \sum_{k\in K^I} c_{ht} g_{kt} \\
&\qquad + \sum_{t\in T} \sum_{p\in P} \sum_{(p,j)\in A^I} c_{bt} f_{pjt} \\
&\sum_{(p,j)\in A^I} f_{pjt} \leq s_{pt} && \forall p \in P, \ \forall t \in T \\
&\sum_{(j,i)\in A^I} f_{jit} - \sum_{(i,j)\in A^I} f_{ijt} + d_{it} \pi 3^*_{it} \geq d_{it} && \forall i \in M, \ \forall t \in T \\
&g_{kt-1} + \sum_{(j,k)\in A^I} f_{jkt} = g_{kt} + \sum_{(k,j)\in A^I} f_{kjt} && \forall k \in K^I, \ \forall t \in T \qquad (3.72) \\
&g_{kt} \leq \overline{C'_k} + M\pi 1^*_{kt} && \forall k \in K^I, \ \forall t \in T \\
&f_{ijt} \leq u + M\pi 2^*_{ijt} && \forall (i,j) \in A^I, \ \forall t \in T \\
&\beta f_{ijt} - \Delta_{ij} \leq 0 && \forall (i,j) \in A^I, \ \forall t \in T
\end{aligned}$$

At each iteration of the problem, an additional optimality cut is generated by relaxing the infeasible sub problem. The strategy determines maximum number of constraints that can yield feasible system from infeasible sub problem. Multi cut generation procedure is applied within Benders Decomposition framework. The MFS cut restrict

the objectify function in case of infeasibility. The general procedure of MFS can be seen in Figure 3.3.

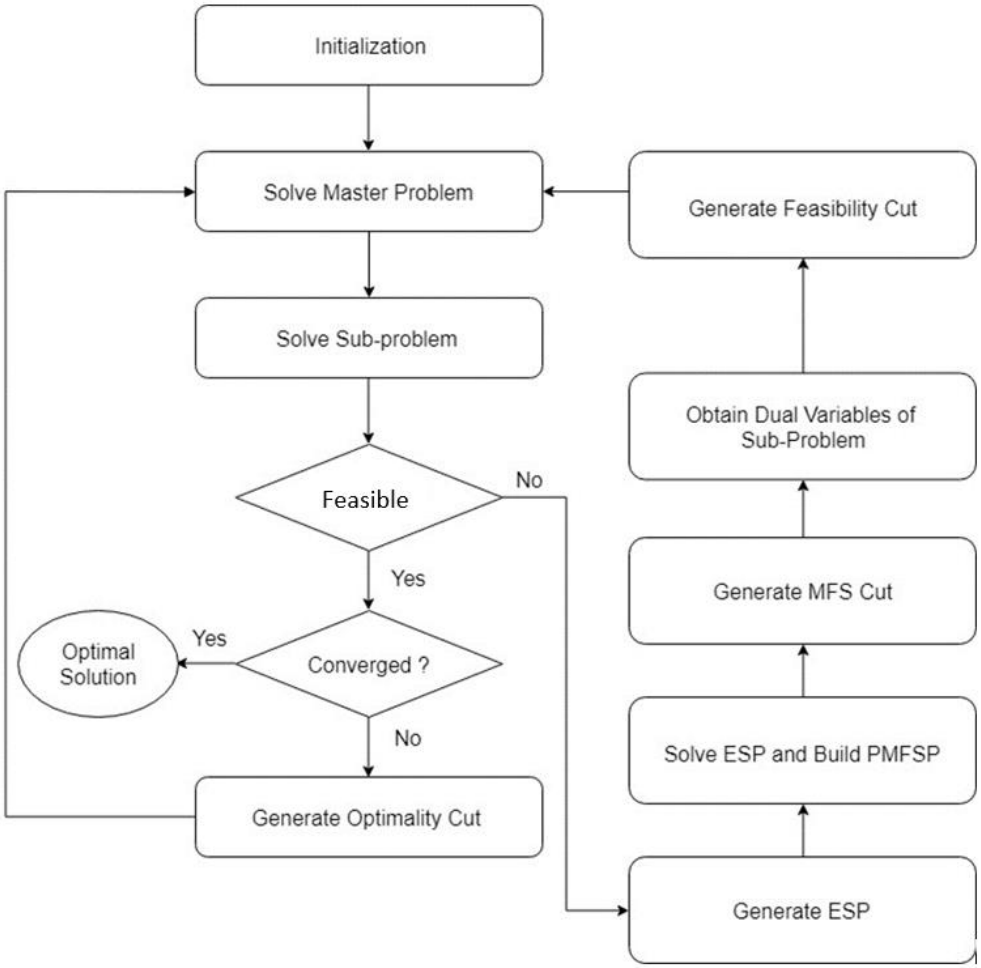

Figure 3.3. Flowchart for the Benders Decomposition with MFS Algorithm

The optimality cut that is added is the following:

$$\eta \geq \sum_{p\in P}\sum_{t\in T} w_{1pt}^{I}\, s_{pt} + \sum_{i\in M}\sum_{t\in T} w_{2it}^{I}\, d_{it} + \sum_{k\in K^{I}}\sum_{t\in T} w_{4kt}^{I}\, \bar{C}_k + \sum_{(i,j)\in A^{I}}\sum_{t\in T} w_{5ijt}^{I}\, u \tag{3.73}$$

In this chapter, deterministic and stochastic version of mathematical model is generated. Solution procedure is applied to the mathematical model. There different Benders decomposition is explained with flow charts of each algorithm.

CHAPTER 4

COMPUTATIONAL EXPERIMENTS

4.1. Deterministic Mathematical Model Experiments

The suggested procedures are tested on a series of randomly generated test problems. The test problems are solved by using three versions of Benders decompositions. These versions are as follows:

b) Benders decomposition with feasibility cuts involving extreme rays
c) Benders decomposition with feasibility cuts involving extreme points of W
d) Benders decomposition with maximum feasible subsystem cut generation

4.1.1. Generation of Test Problems

The test problems are generated taking into consideration three aspects as follows:

b) The structure of the network,
c) The supply and demand values of the nodes,
d) The cost parameters.

4.1.2. The Structure of Network

The size of the test problems is determined by the number of demand nodes, n, the number of supply points, p, and the number of possible storage sites, k. In our experimentation, n is taken as 30, 40, and 50; p is taken between 1 and 3; and k is taken as 2 and 4. The coordinates of the supply, demand nodes and the storage sites are determined randomly within a 50×50 grid structure.

One structure is used for the location of ports such that they are located near the edges of the grid. Rounded Euclidean distances between the nodes are calculated, and the

arcs of the complete graph are sorted according to the Euclidean distances in a non-decreasing manner.

The number of arcs, *m*, in the generated test problems are determined by considering three values for ratios of *m/n* as 3, 5 and 6. The arcs are included in the generated networks using a two-step procedure. In the first step, to guarantee connectivity, arcs *(i,j)* are created such that $j=i+1$, i.e. each node *i* is connected to node $i+1$. In the second step, the remaining arcs are selected from the sorted list, taking the ones with the smallest Euclidean distances until m arcs are selected.

With 3 possible values for *n*, 2 possible values for *p*, 2 possible values of *k*, and 3 possible values for *m/n* ratio, 36 different sets of test problems are created. For each set, five instances are generated.

4.1.3. The Supply and Demand Values of Nodes

The monthly demands of the demand nodes are generated. The monthly demand values of cities have seasonal trends so that there exist high consumption in winter months and less consumption in summer.

The demand data generation process is divided into two parts depending on the season of the year. The winter demand terms include the months between 11th month and 4th month of the year. The demand is generated by using uniform distribution between (21, 540). Whereas, the summer periods include months between 5th and 10th months of year. The uniform distribution between (2, 360) is used. Demand data is generated by analyzing Turkish cities demand in 2016 and 2017.

The supply values are generated to meet all the observed demand so that infeasibility will not be observed in the solutions. The infeasibility in the problem can occur as a result of both unconnected nodes and unmet demands. As a result, infeasibility related with insufficient supply amounts will be resolved. Storage units are used as buffer in case of insufficient supply at the upcoming time periods. Gas will be stored in excess

conditions so that it will be used in case of scarcity. Moreover, ınfeasibility due to unconnected nodes will be solved by feasibility cuts in benders decomposition.

The total required supply is divided into number of ports in the problem structure with predefined ratios. In case of three ports case, the supply ratios of ports are 50%, 25% and 25%. These rations will make system to depend on specific ports. Therefore, the characteristics of real system will be observed.

4.1.4. The Cost Parameters

Pipe establishment and pipe diameter cost terms are determined after pre-test of mathematical model with different values of these parameters. The parameter sets that constitute flexible network design are selected. The flexible network means that network tree has consist of several branches with lots of sub trees. Indeed, pipeline network doesn't depend on one single pipeline segment. Because these kind of systems are more open to vulnerabilities such that ın case of damage at the beginning these pipeline networks, lots of downstream nodes will be affected by these damages.

In the literature, there doesn't exist any study that combines pipeline network design with storage units. Therefore, the cost term estimate of storage units will be based on models preliminary runs. The correct cost term for storage units are determined by analyzing the mathematical model. The establishing cost of storage units are calculated in terms of yearly equivalent in the model. Storage opening is assumed as 1000 $ and storage holding cost is used as 0.3 ($/MMscfd/month). Demands are divided into 2 different periods for high and low season which consists of 6 month periods. At the same time, mathematical model is solved for 12 month periods so that it will be feasible to hold the gas in the storage at least 5 months. Sale and purchase cost are included in the model in terms of total rate that are imported from entrance node and served to customers. These costs are related with demand qualities in terms of volume. The related cost terms can be seen in Table 4.1.

Table 4.1. *Cost Parameters*

Parameters	Value
Storage Opening (c_S)	1000 $/month
Pipe Establishment (c_o)	100 $/km-month
Pipe Diameter (c_p)	1 $/month-inch-km
Storage Holding (c_{ht})	0.3 $/MMscfd
Purchase Cost (c_{bt})	1 $/MMscfd

Storage unit capacity is used as 500 MMscfd which is equal to ten percent of total demand in 12-month time periods. Therefore, in case of sudden breakdowns, system can enable to meet 10% of demand. Flow limit parameter equals to highest supply amount from ports. Model parameters can be seen in Table 4.2.

Table 4.2. *Model Parameters*

Parameters	Value
d_{it}	Unif(21,540) and Unif(2,360)
s_{pt}	Equal to summation of total demand in periods
γ_{ij}	Euclidian distance between nodes
$\overline{C_k}$:	500 MMscfd
u :	$max\{s_{pt}\}$

We first test the computational performance of PSNDP, PSNDP-B1, PSNDP-B2 and PSNDP-B3. In comparing the models, one of the performance measures is the solution time in CPU seconds. A time limit of 10800 CPU seconds is set for all the models. The numerical experiments are conducted in computer with Intel Core i5 1.60 GHz and 8GB RAM. The example model sets can be seen in Table 4.3.

Table 4.3. *Example Model Set*

Set No	*Demand Nodes*	*Supply Nodes*	*Storage Nodes*	*Number of Arcs*
1-1	30	1	2	90
1-2	30	1	2	120
1-3	30	1	2	150
1-4	30	1	4	90
1-5	30	1	4	120
1-6	30	1	4	150
1-7	30	3	2	90
1-8	30	3	2	120
1-9	30	3	2	150
1-10	30	3	4	90
1-11	30	3	4	120
1-12	30	3	4	150
2-1	40	1	2	120
2-2	40	1	2	160
2-3	40	1	2	200
2-4	40	1	4	120
2-5	40	1	4	160
2-6	40	1	4	200
2-7	40	3	2	120
2-8	40	3	2	160
2-9	40	3	2	200
2-10	40	3	4	120
2-11	40	3	4	160
2-12	40	3	4	200
3-1	50	1	2	150
3-2	50	1	2	200
3-3	50	1	2	250
3-4	50	1	4	150
3-5	50	1	4	200

3-6	50	1	4	250
3-7	50	3	2	150
3-8	50	3	2	200
3-9	50	3	2	250
3-10	50	3	4	150
3-11	50	3	4	200
3-12	50	3	4	250

4.1.5. Deterministic Mathematical Model Solution Results

The proposed mathematical model determines pipeline length and diameters. The proposed solution techniques for the randomly generated network is applied to yield optimal configuration of pipeline network with optimal diameter and length. In the randomly generated network, there exist one or three entrance point for the pipeline system. The model is solved for different scenarios of supply and demand values. These scenarios are generated by analyzing real supply and demand values of big cities.

The scenarios are generated for the time horizon of 12 months. The scenarios are generated for supply and demand values of entrance and demand nodes. The time horizon is divided into 2 periods such as low and high season which is adapted from real pipeline system. There exists high natural gas usage in winter periods due to heating purposes. The usage amounts decrease toward the summer months. The demand values are generated for 30 and 50 node cases by using uniform distribution. The two different uniform distribution is used to generate demand values of each node during time periods.

Each time period is divided into 2 parts so that there will be high level of supply with respect to demand values in the first period. In the second period, supply values will be lower than demand values. There will be three months to store natural gas in the storages in each time periods. Infeasibility in the model solutions are eliminated by this strategy.

The two stage stochastic mathematical model is solved by using benders decomposition. Three different benders decomposition is proposed to compare the solutions methods. The mathematical model is solved as single stage problem with GAMS cplex solver so that there exists way to compare the solution performances of benders decomposition versions. All computational experiments are carried out on Gams version 23.5.1.

We have solved 5 problem instance for each replications 12 different scenarios are tested with the mathematical model. These instances are used to qualify the solution performance of proposed methods. The objective of proposed study is to decrease operational cost of designing pipeline system. The problem elements are supply, demands nodes and pipeline length and diameters. The related cost terms of instance are generated based on the engineering economy concepts by assuming one year working period of each element. Storage cost of natural gas is generated by assuming a fraction of sale price of gas to demand nodes. The sale and purchase price of natural gas is created by concerning enough time periods to store the natural gas. The candidate location for supply point is selected around outline of data points. Pipeline length are calculated by Euclidian distance between nodes.

4.1.6. Performance of Solution Methods

In this part of study, the solution methods are compared with each other. The stopping conditions are 3-hour time limit, 0.00% optimality gap. In addition, the benders decomposition iterations continue until upper bound a lower bound converge to each other. Convergence parameter for benders decomposition is selected as 200 after preliminary runs. 3 different benders decomposition as described in the previous sections and single stage methods are applied to solve mathematical model. Solution results can be analyzed in Table 4.4, 4.5 and 4.6. Model results of individual runs are available in Appendix-A.

Performance of solution method is compared based on the solution time and number of optimal solution found in each replications. Moreover, total number of iterations done in each replication is also reported.

Single stage solution is considered as pilot group in order to qualify the alternative solution methods. It is firstly investigated the solution performance of bender decomposition modifications under supply and demand scenarios.

After conducting related evaluations, first insight is that application of benders decomposition provides a remarkable enhancement over single stage solution based on solution times. Single stage solution cannot reach to defined optimality gap within 3-hour time limits for all cases. Average solution time equals to 10800 CPU seconds of the single stage solution. For the 30 node scenarios, PSNDP-B1 solves the instance in higher CPU seconds with respect to PSNDP-B2. It is actually expecting that PSNDP-B2 needs more CPU seconds to solve the models because PSNDP-B2 case also solves another problem and generates extra cuts to reach optimal solution. However, PSNDP-B2 case yields optimal solution in smaller time intervals. It is also observed that solution time increases toward case number 12 due to increase in the problem size for all solution methods as it is expected. PSNDP-B3 case needs higher CPU seconds in all cases.

Gap column in the solution table show the average optimality gap of replications for each case. As one of the termination criteria is to reach zero percent optimality gap, each solution strategy reaches to zero percent optimality gap within defined time interval. The number of iterations in benders decomposition increase as problem size increases. The number of arcs in pipeline network yields to increase the solution times.

Table 4.4. *Deterministic Model Results*

	PSNDP			PSNDP-B1			
Set No	CPU (s)	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
1-1	10800	3.856	0	1565.400	2.4	0	5
1-2	10800	5.074	0	1678.6	4	0	5
1-3	10800	5.074	0	2073	3	0	5
1-4	10800	4.68	0	1604.8	4	0	5
1-5	10800	7.336	0	1847.6	4.2	0	5
1-6	10800	11.362	0	2133	5	0	5
1-7	10800	5.104	0	1644.6	5.2	0	5
1-8	10800	7.792	0	1990.6	5	0	5
1-9	10800	11.254	0	2243	5.4	0	5
1-10	10800	6.848	0	1845.6	5	0	5
1-11	10800	10.232	0	2475.4	5	0	5
1-12	10800	12.504	0	2560.4	5	0	5

	PSNDP-B2				PSNDP-B3			
Set No	CPU (s)	# iter.	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
1-1	1010	3.4	0	5	1689.8	3	0	5
1-2	1286.4	3.6	0	5	1743.2	5.2	0	5
1-3	1384.8	3.2	0	5	2207.6	4.6	0	5
1-4	1117.2	4.4	0	5	1923.4	5.2	0	5
1-5	1252.8	4	0	5	2262.2	5	0	5
1-6	1345.2	3.4	0	5	2345.2	5	0	5
1-7	1157.2	4	0	5	2455.2	6	0	5
1-8	1327.8	5	0	5	2465.8	5.8	0	5
1-9	2465.8	5.6	0	5	2606.6	5.8	0	5
1-10	1318.6	5	0	5	2497	5	0	5
1-11	2212.2	5	0	5	2510.8	5	0	5
1-12	2308.8	5	0	5	1689.8	3	0	5

Table 4.5. *Deterministic Model Results*

	PSNDP			PSNDP-B1			
Set No	CPU (s)	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
2-1	10800	8.8	0	4210.6	3	0	5
2-2	10800	10.2	0	4418.2	4	0	5
2-3	10800	11.1	0	4900.6	4	0	5
2-4	10800	8.9	0	4584.8	5	0	5
2-5	10800	11.5	0	4606.8	4	0	5
2-6	10800	12.8	0	5179.0	4.2	0	5
2-7	10800	6.9	0	4896.8	4	0	5
2-8	10800	10.3	0	5203.2	4	0	5
2-9	10800	13.7	0	5470.0	4	0	5
2-10	10800	7.1	0	5229.8	4.2	0	5
2-11	10800	10.9	0	5776.0	4	0	5
2-12	10800	14.5	0	5968.0	4.6	0	5

	PSNDP-B2				PSNDP-B3			
Set No	CPU (s)	# iter.	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
2-1	4000.0	3	0	5	4802.6	4	0	5
2-2	4203.0	3.8	0	5	4949	4	0	5
2-3	4694.0	5	0	5	5125.2	4	0	5
2-4	4347.4	4.4	0	5	4714.6	4.6	0	5
2-5	4416.8	4.2	0	5	4906.4	4	0	5
2-6	4790.4	5	0	5	5279.6	4	0	5
2-7	4782.4	5.6	0	5	5022.2	4.2	0	5
2-8	5470.6	4.4	0	5	5642.8	4.2	0	5
2-9	5913.6	4.6	0	5	6049	4.2	0	5
2-10	4905.8	5.2	0	5	5601.4	4	0	5
2-11	5484.4	5	0	5	5977.8	4.2	0	5
2-12	5806.0	5.6	0	5	6045.8	4.6	0	5

Table 4.6. *Deterministic Model Results*

	PSNDP			PSNDP-B1			
Set No	CPU (s)	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
3-1	10800	6.4	0	5079.6	3.2	0	5
3-2	10800	9.1	0	5636.4	3.6	0	5
3-3	10800	13.7	0	6060.6	4	0	5
3-4	10800	8.5	0	5131.6	4	0	5
3-5	10800	12.2	0	5609.4	4	0	5
3-6	10800	13.1	0	6095.6	4.2	0	5
3-7	10800	7.5	0	5082.4	4	0	5
3-8	10800	10.9	0	5701.2	4	0	5
3-9	10800	14.4	0	6092	4.2	0	5
3-10	10800	7.3	0	5127.8	4.2	0	5
3-11	10800	11.4	0	5703.4	4.4	0	5
3-12	10800	14.6	0	6108.8	4.4	0	5

	PSNDP-B2				PSNDP-B3			
Set No	CPU (s)	# iter.	Gap (%)	Opt.	CPU (s)	# iter.	Gap (%)	Opt.
3-1	4764.8	3	0	5	5653.6	4	0	5
3-2	5349.8	3.4	0	5	6275	4.2	0	5
3-3	5836	4	0	5	6706.6	4.4	0	5
3-4	4885.2	4	0	5	5293	4.2	0	5
3-5	5350	3.8	0	5	5851.4	4.4	0	5
3-6	5880.2	4	0	5	6241.6	4.4	0	5
3-7	4800.4	4	0	5	5608	4.4	0	5
3-8	5493.8	4	0	5	6423.6	4.6	0	5
3-9	5903.8	4	0	5	6761.6	4.6	0	5
3-10	4937	4	0	5	5857.8	4.8	0	5
3-11	5438	4	0	5	6378.4	4.8	0	5
3-12	5859.2	4	0	5	6913.8	5	0	5

4.2. Stochastic Mathematical Model Experiments

We formulate a two stage stochastic mixed integer linear programing model to determine arcs, flow amounts between nodes and locations of storage facilities.

Capacities of storage nodes and monthly levels are determined to minimize the expected operation cost.

In this part, we extend the deterministic mathematical model by including uncertainty in natural gas demand. Storage capacities and flow amounts between network nodes will be determined by the model. In the two stage stochastic mathematical model, first stage decisions determine optimal network design with storage nodes and pipes between demand nodes. The second stage addresses the minimization of expected flow cost over a set of scenarios that capture demand uncertainty. We consider to test our solution methods on stochastic programing.

4.2.1. Generation of Test Problems

We generated two problem classes with different sizes. Detailed information about the problem classes are given in Table 4.7. Our problem instances are generated based on the deterministic problems case and Turkish pipeline design problem. It is observed that problem size time limit of 3 hours will be reached with problems that include more than 40 demand nodes.

Table 4.7. *Problem Classes*

Set No	Demand Nodes	Supply Nodes	Storage Nodes	# of Arcs	# of Scenarios
C1-1	30	1	2	90	27
C1-2	30	1	2	120	27
C1-3	30	1	2	150	27
C1-4	30	1	4	90	27
C1-5	30	1	4	120	27
C1-6	30	1	4	150	27

C1-7	30	3	2	90	27
C1-8	30	3	2	120	27
C1-9	30	3	2	150	27
C1-10	30	3	4	90	27
C1-11	30	3	4	120	27
C1-12	30	3	4	150	27

4.2.2. The Structure of Network

Objective of the model will be minimization of expected cost. Our cost parameters are used as the ones in the deterministic case. Candidate storage locations are generated randomly. Euclidean distances between nodes are used. In terms of scenarios, only demands are included as uncertain parameter. Demand values are generated by using randomly and independently uniform distribution. Three different values such as low, medium and high demand values are used. Related parameters can be seen in Table 4.8. The node set divided into 3 equal parts. For each part, three different demand levels such as low, medium and high are used. Therefore, 27 scenarios are generated. The probability of each scenarios equals to each other. Demand values are determined based on deterministic case.

4.2.3. Model Parameters

As natural gas demand follows seasonal trend the deterministic natural gas demand values divided for three equal intervals as low, medium and high. Total supply amounts are equated to total demand at the end of time periods. Storage units' capacity is used 1000 MMscfd.

Table 4.8. *Model Parameters*

Parameters	
d_{itw} (Low) t=1,..,6	Unif(20,40)
d_{itw} (Medium) t=1,..,6	Unif(60,80)
d_{itw} (High) t=1,..,6	Unif(100,120)
d_{itw} (Low) t=7,..,12	Unif(180,200)
d_{itw} (Medium) t=7,..,12	Unif(220,240)
d_{itw} (High) t=7,..,12	Unif(260,280)
$\overline{c_k}$	1000 MMscfd
s_{ptw}	$\sum_{\forall p \in P} \sum_{\forall t \in T} s_{ptw} = \sum_{\forall i \in N} \sum_{\forall t \in T} d_{itw} \ \forall w$

4.2.4. Scenario Generation

In terms of uncertainty representation, we generate scenarios in order to represent population. The model tested with fixed number of scenarios so that generated scenarios covers all uncertainties. The problem sizes are reduced to solve the models in reasonable time with proposed solution method. Scenario generation process firstly starts with dividing the node set into three equal cluster based on their proximity to each other. Example clusters and their centroids can be seen in Figure 4.1. Node clusters are used to assign demand values. For each cluster, three different demand levels are generated. Scenarios are classified as low, medium or high demand with equal probability. Uniform distribution is used to generate demand parameters for low, medium and high demand values. In our scenario generation process, selected scenarios are assigned to each node in the cluster.

It is required to obtained enough scenarios in order to solve corresponding stochastics program within given set. The scenario should be the good representation of underlying distribution. Therefore, stability and bias requirements should be meet. Due to the size and complexity of our problem, it is difficult to verify bias and stability requirements.

We defined ϕ as the set of all possible scenarios of the uncertain parameters in the model and $w \in \phi$ represent for a particular scenario.

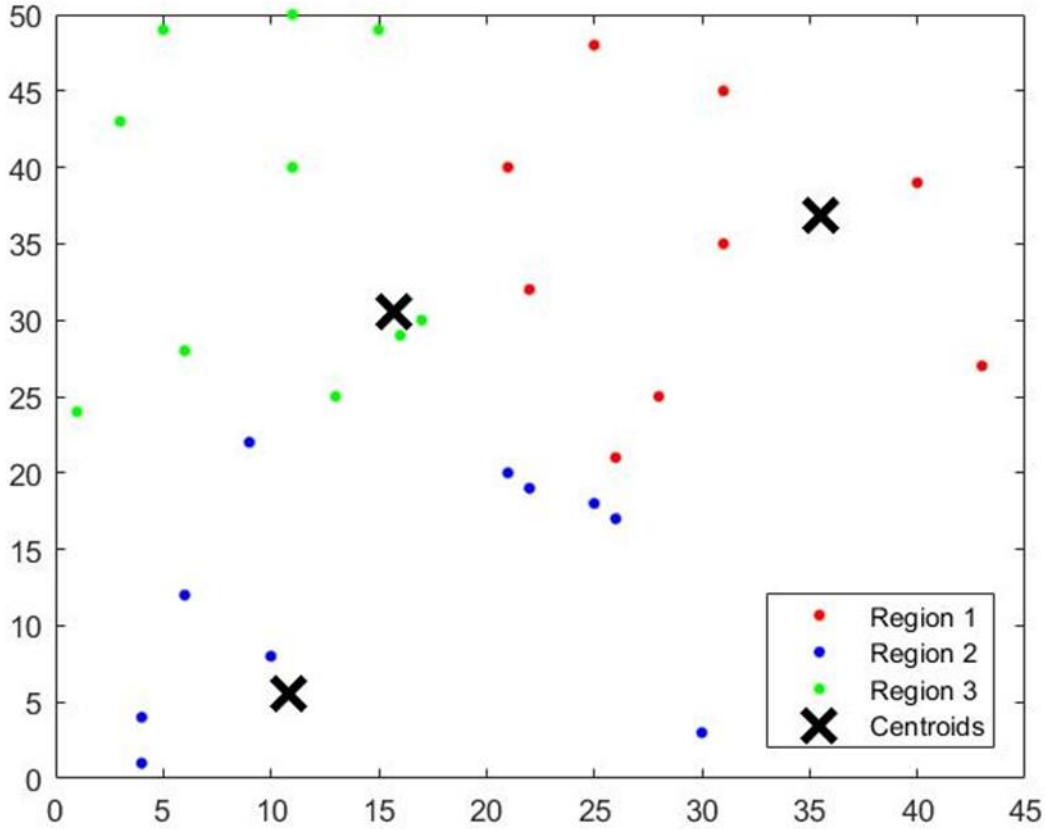

Figure 4.1. Node Clusters and Centroids

4.2.5. Solution Method

In deterministic problem case, it is observed that benders decomposition methods provide significant improvements in terms of solution times. After comparing the

solution performances of PSNDP-B1, PSNDP-B2 and PSNDP-B3, second solution method performs better than others in terms of solution times and total number of iterations. Analysis within each solution strategy show that PSNDP-B1 and PSNDP-B3 increases solution times and degrades computation performance so that higher computational times required. Therefore, we eliminate the first and third solution methods from further considerations.

Our solution approach PSNDP-B2 is relatively easy to implement to solve two stage stochastic programing. It is also common to use Benders decomposition methods for integer first stage variables and continues second stage variables as in our case. Our termination criteria are 0% optimality gap or 3-hour time limit. Comparison of performances are obtained by evaluating average and worst case solution times. Number of unsolved problems and gaps are also reported.

4.2.6. Performance of Solution Methods

Within 3-hour time limit and 0% percent optimality gap conditions, PSNDP-B2 is used to solve two stage stochastic mathematical model. For each set, 5 replications are solved. Model results can be seen in Table 4.9. Individual model results and network costs are available in Appendix-B. Sets number between 1 and 5 are solved optimality in all replications. 5 problem set cannot be solved in any replication.

Optimality gap values are also analyzed for unsolved instances. The average of optimality gaps is increased toward set number 12. It is observed that as problem size increases, the optimality gap also increases for the unsolved instances. Number of iterations are higher for the stochastic problems with respect to deterministic case. For the one port cases, mathematical model choses to have pipeline between closest high demand city. After that, this node is used for hub location to disturbing natural gas. Therefore, model have to generate one big size pipeline for network so that remaining gas is distributed to system by smaller pipelines.

For three port cases, model generally choses to have more pipeline with respect to one port cases. Especially, model generate pipeline between ports and the nodes around

them. This kind of structure can be results of variation in nodes demands. Rather than generating one big pipeline, natural gas is distributed more uniformly with respect one port cases. At the same time, networks of three port cases are generated with more storage units.

It is also important that model generally choses to open storage units which are around the cities that has demand values higher that system's average or the ones that are close to entrance points. By constructing storage units around the entrance points, model distributed the natural gas throughout storage nodes so that there is no need to construct additional pipeline to storage nodes.

Table 4.9. *Stochastic Model Results*

Set No	CPU (s)	# of Iteration	Gap (%)	Opt.
C1	10353,00	8	0	5
C2	10392,60	8,6	0	5
C3	10672,80	9,4	0	5
C4	10540,00	8,8	0	5
C5	10552,60	9	0	5
C6	10800	0	1,874	0
C7	10465,00	8,6	2,34	3
C8	10612,40	9,5	2,65	2
C9	10800	0	3,724	0
C10	10800	0	3,506	0
C11	10800	0	4,142	0
C12	10800	0	5,112	0

CHAPTER 5

TURKISH PIPELINE SYSTEM CASE STUDY

A Natural gas is one of the main factors determining the political, economic and social status of Turkey. Turkish government changed the important structures of the natural gas sector. First natural gas import agreement was signed in 1986 between BOTAŞ and Soyuzgazexport company for 25 years [21]. Natural gas was imported first time in 1987. The following year, LNG purchase agreement was signed with Algeria. The first LNG facility started to operate in 1994. After that, natural gas agreements with Iran and Russian was signed. Both agreements were signed for a period of 25 years. A 15-year period natural gas agreement with Azerbaijan was signed in 2001. At the same year, the law on natural gas market (No 4646) was constructed, striding a significant step in the liberalization of Natural Gas Market of Turkey. Natural gas distribution was launched initially in Ankara in 1988. Later in 1992 Istanbul and Bursa switched to natural gas use, and Eskişehir and Kocaeli followed suit in 1996. There exist current construction projects for the pipeline and storage system. Pipeline projects that are in the agenda of government are as below [39].

- Russian-Turkey Natural Gas Pipeline (West Line)
- Blue Stream Gas Pipeline
- Eastern Anatolian Natural Gas Main Transmission Pipeline (Iran - Turkey)
- Baku – Tbilisi – Erzurum Natural Gas Pipeline
- Turkey – Greece Natural Gas Interconnection
- Trans – Anatolia Natural Gas Pipeline Project
- Turkish Stream Gas Pipeline

Increasing natural gas demand lead to optimize pipeline network system of Turkey. The geographic location of Turkey creates a bridge between natural gas suppliers and

consumers. Turkey has a strategical role for providing uninterruptable natural gas for its own demand and downstream countries in the pipeline system.

Turkey pipeline system is controlled by BOTAŞ which is Turkish Petroleum Pipeline Corporation. BOTAŞ is carrying out its business activities as the most important market player based on its licenses issued by the Energy Market Regulatory Authority for import, export, transportation and storage of natural gas. BOTAŞ wants to provide storage capacity of 10% of imported gas.

Turkey currently imports most of its natural gas from Russia, Iran and Azerbaijan. Currently, there exist 4 entrance points of natural gas pipeline. These are Malkoçlar, Durusu, Türkgözü and Gürbulak. Two new international pipeline projects- TANAP and TurkStream- will generate 3 new entrance points. TANAP will carry Azerbaijani gas to Europe. It will deliver gas at Eskisehir and Trakya. TurkStream project will deliver Russian gas at Kıyıköy. Trans Anatolian Natural Gas Pipeline (TANAP) will provide stronger economic and political relationship with European Union. It will also provide to secure and diversify supply by carrying Azerbaijan natural gas. Eskişehir and Thrace are 2 exit points of pipeline in Turkey [51].

The pipeline entrance points and amount can be seen in the Figure 5.1 and Table 5.1. The Azerbaijan gas enters the country from Türkgözü which has the capacity of 19.1 mcm/day. Gürbulak delivers the Iranian gas with capacity of 28.6 mcm/day. Durusu and Malkoçlar carry the Russian gas with corresponding capacities of 47.3 and 51.4 mcm/day. Detailed natural gas imports can be seen in Table 5.1. Most of the new entrance points will start to operate after 2020. Malkoçlar and Kıyıköy entrance points will operate with highest amount of natural gas imported. After 2020, only decrement in natural gas amount will be observed in Malkoçlar gate due to Kıyıköy station.

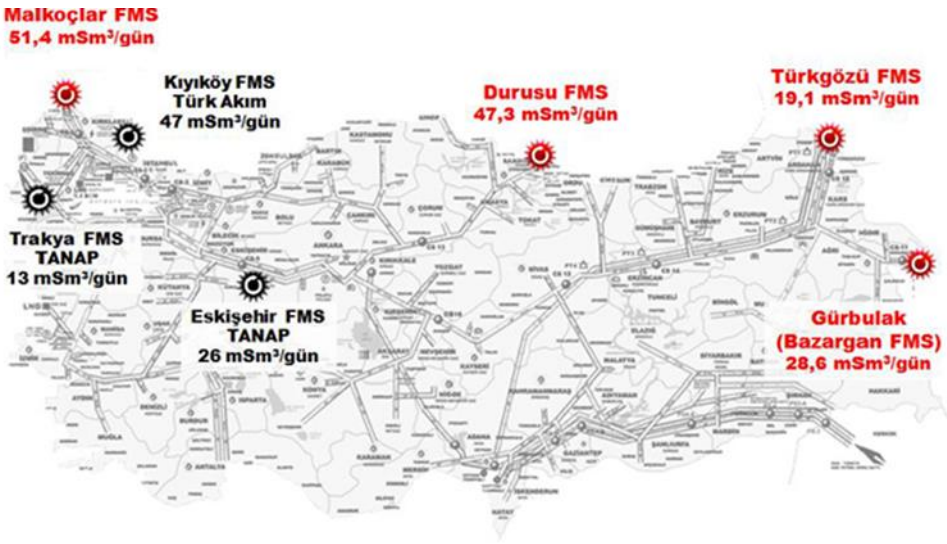

Figure 5.1. Natural gas pipeline points [20]

Table 5.1. *Natural Gas Imports mcm/day [20]*

Entry Points	2017	2018	2019	2020	2021	2022	2023
Malkoçlar	51.4	51.4	51.4	45.1	45.1	45.1	45.1
Durusu	47.3	47.3	47.3	47.3	47.3	47.3	47.3
Gürbulak	28.6	28.6	28.6	28.6	28.6	28.6	28.6
Türkgözü	19.1	19.1	19.1	19.1	19.1	19.1	19.1
Eskisehir	0	5.7	11.3	14	26	26	26
Trakya	0	0	0	0	13	13	13
Kıyıköy	0	0	0	47	47	47	47

Turkey also uses Liquefied Natural Gas (LNG). The cost of LNG is higher than pipeline gas. The lead time and logistic related problems also make pipeline and storage facilities more favorable. There exist 4 LNG terminals in Turkey. LNG

terminals provides the diversification of supply resources. LNG imports are equal to nearly 17 percent of total imports in 2017. The expected amount can be seen in below table. The LNG entrance points can be seen in the Figure 5.2. Detailed LNG imports can be seen in Table 5.2. Total LNG import increases up to 2023. Saros FSRU started to operate at 2018.

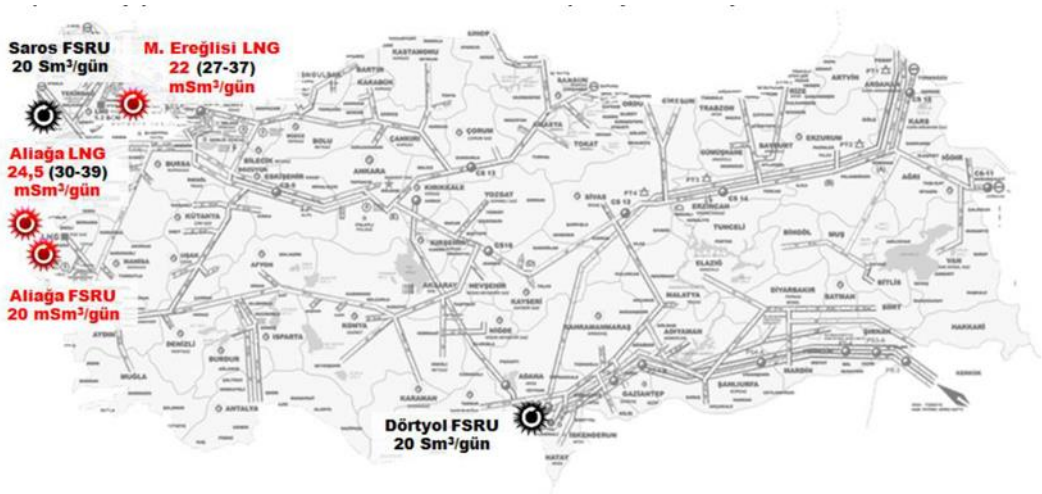

Figure 5.2. LNG entrance points [20]

Table 5.2. *LNG Imports [51]*

Entry Points	2017	2018	2019	2020	2021	2022	2023
M. Ereğlisi	22	37	37	37	37	37	37
Aliağa LNG	24.5	30.9	30.9	30.9	39	39	39
Aliağa FSRU	20	14.1	14.1	14.1	20	20	20
Saros FSRU	0	20	20	20	20	20	20

Turkey has been depending on natural gas import for many years. Moreover, this dependence to imports has become higher in recent years. Turkey has increased natural gas consumption since 1980s. According to EPDK, Turkey used 0.5 billion cubic meter of gas at 1987 and this usage has increased to 35 billion cubic meter at 2007. Total amount of 52 billion cubic meters of gas is used in 2017. The natural gas usage by different customers in Turkey is given in Figure. 5.3.

According to Figure 5.3, household natural gas demand and industrial demand is expected to increase steadily. Power plant demand will start to decrease after 2019 [21]. The graph explains total natural gas demand in terms of billion cubic meter.

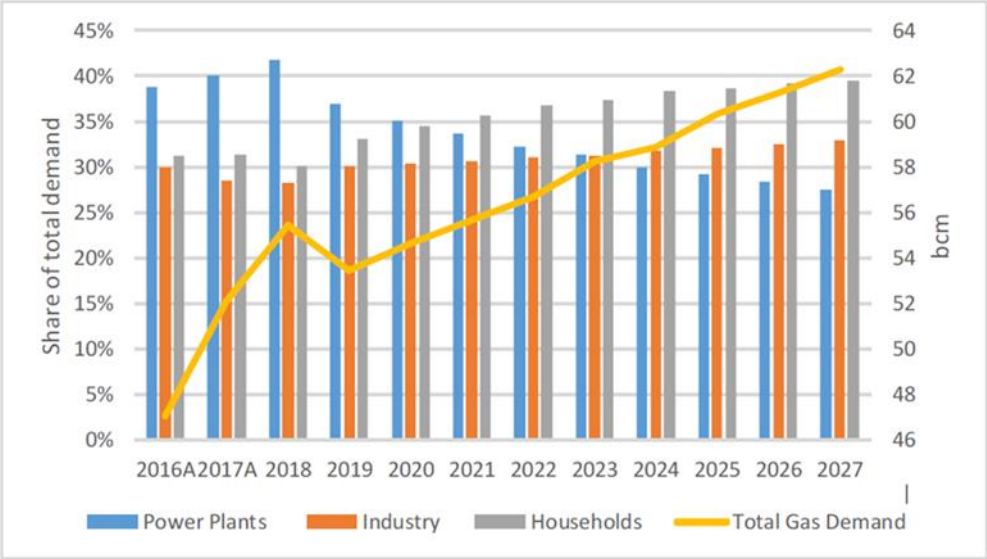

Figure 5.3. Natural gas usage by customer type between 2016 and 2027 [51]

There exist huge demand fluctuations throughout months. Moreover, BOTAŞ can reach to its peak delivery levels at winter months. According to EPDK, the monthly peak delivery levels are in Table 5.3 in terms of MSm3. BOTAŞ states that send out capacity of the pipeline system is 200 mcm/day. Therefore, it is possible that the pipeline system can generate bottlenecks to meet natural gas demand at winter months.

Table 5.3. *Monthly Peak Delivery Levels in 2017 [20]*

2017	01	02	03	04	05	06
	232,88	243,01	196,40	157,14	130,94	132,66
	07	08	09	10	11	12
	139,06	136,63	134,41	171,05	213,06	229,33

Energy Regulatory Office of Turkey (EPDK) published the natural gas imports based on entrance points for 2017 as given in Table 5.4. Monthly imported amounts between 2013 and 2017 is given in Table 5.5 and Figure 5.4.

Table 5.4. *Imported Natural Gas in 2017 MSm3 [20]*

Enterance Point	Imported Gas	Percentage
Durusu	15.584,85	28,21
Malkoçlar	13.104,78	23,2
Gürbulak	9.250,64	16,74
Türkgözü	6.544,40	11,85
Egegaz Aliağa LNG	4.235,28	7,67
Marmara Eereğlisi LNG	3.676,33	6,65
ETKİ Aliağa LNG	2.744,23	4,97
Dörtyol FSRU	109,44	0,20
Total	55.248,95	100

Table 5.5. *Monthly natural gas imports between 2013 and 2017 [39]*

Months	2013	2014	2015	2016	2017
January	4.536,21	4.791,87	4.997,80	5.392,94	5.914,34
February	4.095,23	4.219,70	6.634,50	3.964,11	5.252,73
March	3.889,33	4.535,44	4.277,50	3.812,55	5.081,27
April	3.447,54	4.114,91	3.572,83	3.553,46	4.250,26
May	3.512,94	3.881,93	3.332,88	3.215,71	3.727,77
June	3.143,21	3.301,27	3.201,40	3.173,14	3.609,89
July	3.404,38	3.689,75	3.325,49	3.220,08	4.135,45
August	3.268,74	3.828,66	3.741,33	3.627,20	4.015,73
September	3.526,97	3.894,04	3.684,66	3.007,63	3.680.77
November	3.545,24	3.442,02	3.745,98	3.456,31	4.137,10
October	3.936.82	4.558,66	4.571,50	4.354,71	5.448,85
December	4.962,36	5.004,02	5.341,20	5.574,35	5.995,79
Total	45.268,97	49.262,27	48.427,07	46.352,17	55.249,95

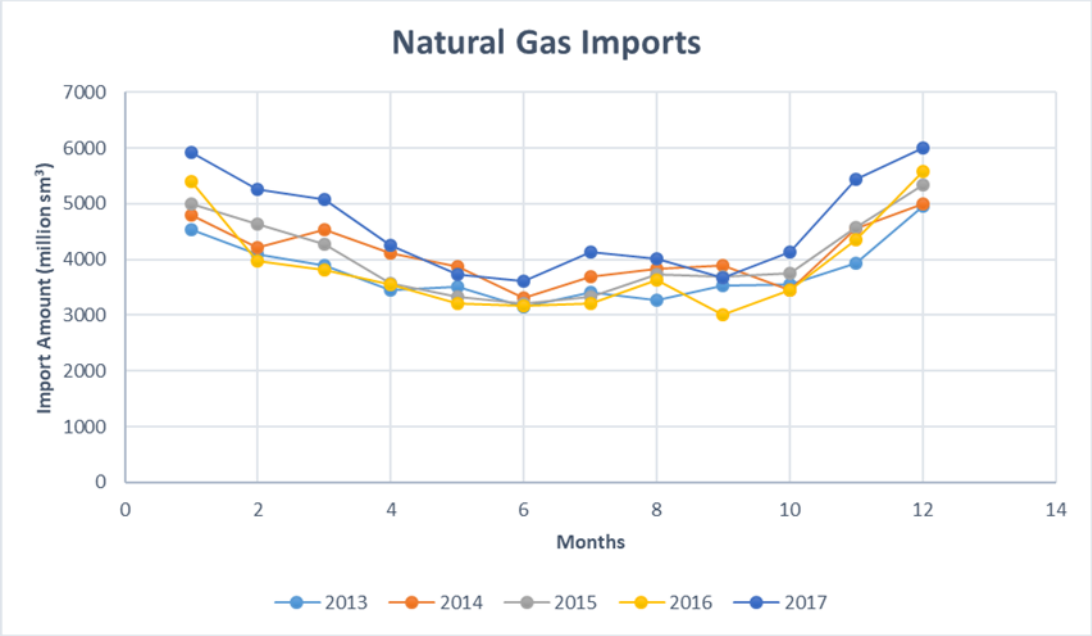

Figure 5.4. Monthly Natural Gas Imports Between 2013 and 2017 [51]

There exists heavy dependence on energy imports that creates energy security problems. Therefore, the storage facilities can partially solve this problem. Storages can provide reliability and stability of gas supply in Turkey. Power generation processes highly uses natural gas due to both environmental concerns and operational flexibilities. In high demand periods, the gas service of power sector can be altered. This leads to an increase in power generation cost because of using oil and coal rather than gas. Energy security can be obtained by meeting the seasonal and daily demand variations. Moreover, Turkey can develop its position as an energy trading center. If the system has enough flexibility and storage capacity, the surplus gas can be exported.

According to the studies on the natural gas supply-demand balances, there is no problem concerning the meeting of the annual natural gas demand. However, during the winter months, when demand is high, the fall of temperatures to levels below seasonal norms, and as a result the rise of consumption to maximum levels, and any faults in the source countries or countries end-route during the same period, can lead to periodic imbalances between supply and demand. Within this framework, the Silivri, Kuzey Marmara and Değirmenköy Natural Gas Storage Facility, which has a

working gas capacity of 2,84 bcm, was taken over by BOTAŞ in order to be used effectively for ensuring seasonal supply- demand balance and supply security.

Turkey has 2 different storage facilities. The Silivri Gas Storage Facility has a capacity of 2.8 bcm which equals to approximately six percent of the annual demand. Tuz Golu Gas storage facility employs 1.2 bcm gas storage capacity. The capacity of Silivri facility is planned to be increased to 4.3 bcm. Underground facilities also create concerns about high initial investment and geological risks. The 20 percent of annual demand is specified as a target level of storage by Turkey's Ministry of Energy and Natural Resources. Within the scope of Kuzey Marmara Natural Gas Storage Expansion Project, developed by the motive of increasing the capacity of related facility, it is planned to increase total storage capacity to 4,6 bcm and withdrawal capacity to 75 mcm/day. Furthermore, the Tuz Gölü Natural Gas Underground Storage Project is planned to reach 5.4 bcm working gas capacity and 80 mcm/day withdrawal capacity by 2023.

Table 5.6. *Storage Units Capacity and Injection Amounts [51]*

		2017	2018	2019	2020	2021	2022	2023
Silivri	Storage(bcm)	2.84	2.84	2.84	2.84	3.34	3.84	4.6
	Injection(mcm/d)	16	16	16	45	45	45	45
Tuzgölü	Storage(bcm)	0.25	0.55	0.55	0.85	2.2	3.4	5.4
	Injection(mcm/d)	15	30	30	30	60	60	60
Total	Storage(bcm)	3.09	3.39	3.39	3.69	5.54	7.24	10
	Injection(mcm/d)	31	46	46	75	105	105	105

5.1. Pipeline System in Turkey

Constructing the natural gas pipeline system started in 1985. With the use of the Natural Gas Usage Survey, which has been done in 1985, the potential natural gas consumption and a feasible route had been determined. The first pipeline system in Turkey was constructed with the help of Russia [42], based on the natural gas demand of 1985. The construction process completed in two phases. Gas entered Turkey from Malkoçlar on the Bulgarian border. In the first phase, the pipeline reached to its first stop, Hamitabat, on June 23, 1987. From this date, in addition to the domestic natural gas, imported natural gas started to be used in electricity production at Trakya Combined Cycle Power Plant in Hamitabat. In the second phase the pipeline reached Ankara in August 1988.The route of the pipeline system is from Hamitabat, Ambarlı, İstanbul, İzmit, Bursa, Eskişehir and Ankara. The total length of the pipeline system was 845 km.

The second pipeline system, which is called as Blue Stream, was constructed under Black Sea from Russia. Entrance point of the pipeline system start from Samsun. Total length of this system is 501 km in Turkey territory. It was completed in 2005. Eastern Anatolian pipeline system carries natural gas from Iran. Starting point of pipeline system is Doğubayazıt to Ankara. Total length of the pipeline system is 1491 km. Another pipeline system is Baku-Tblisi- Erzurum (BTE) system which carries Azerbaijan gas. Total length of the system was 980 km in 2007. Expansion of BTE is currently under construction. Two different projects are under construction in order to meet the rising natural gas demand in the country.

The natural gas pipelines between Turkey and other countries can be seen in Figure 5.5. In the figure, thick red lines show the existing international pipelines and national pipeline inside the country. Green dot lines show planed pipelines to transmit natural gas.

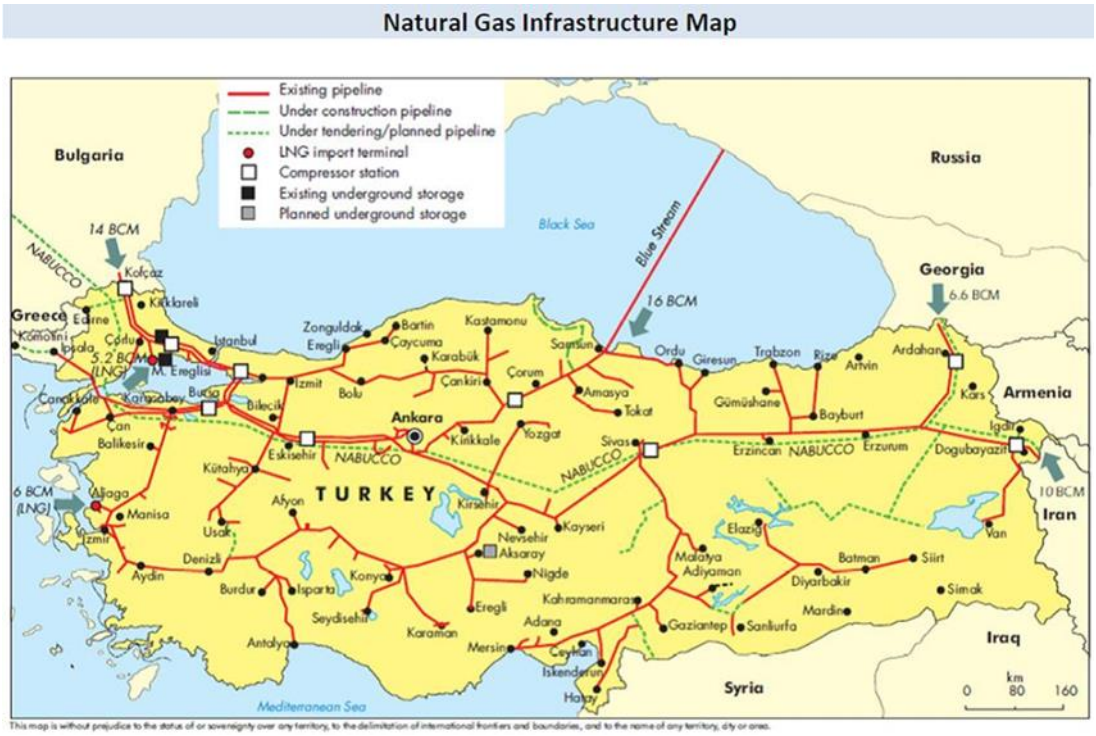

Figure 5.5. Pipeline System in Turkey [53]

Iraq and Turkey pipeline system consists of two parallel pipeline systems. There exists restricted transfer in these pipeline systems due to political concerns. Batman and Dörtyol pipeline system was constructed to transfer the natural gas that is produced in the Batman region. Ceyhan and Kırıkkale pipeline system was constructed to meet natural gas demand of cities interior zones. Russia Malkoç pipeline system carries natural gas up to Ankara natural gas control center. These system is reconstructed to transfer natural gas to Karadeniz and Bursa region. Iran pipeline system is constructed to diversify the natural gas supply. Bursa and İzmir natural gas pipeline system is completed at 2003 to supply natural gas to the Aegean region.

In the next section, we will construct the pipeline system for Turkey from scratch, and compare the results with the current network.

5.2. Deterministic Turkish Pipeline Case Study

The monthly natural gas consumption of the Turkish cities is available in the monthly natural gas report of Turkish Energy Regulatory Authority webpage [20]. PSNDP-B2 is used to solve the Turkish pipeline system model. Case study is designed by concerning three aspects as follows:

a) The structure of the network
b) The supply and demand values of nodes
c) Cost parameters

5.2.1. The Structure of the Network

Distances between nodes are assumed to be the Euclidean distances. The cities are only connected with its neighbor cities.

5.2.2. The Supply and Demand Values of Nodes

Supply values are generated based on total imports given in the above discussion. Supply values used in the model can be seen in Table 5.4. Storage units are selected based on existing storage units in Turkey. Additional storage units in the mathematical model are selected from the existing LNG facilities in Turkey.

Monthly natural gas demand values for 2017 and 2018 of 80 Turkish cities are obtained from EPDK yearly reports. Demand values of regions of Turkey can be seen in Table 5.7. The demand values of İstanbul, Ankara and İzmir yields the 30% of total demand. The percentage of total demand of regions can be seen in below Table 5.7.

Table 5.7. *Demand Values of Regions in Turkey*

	2017	2018
Biggest 3 Cities	32,22 %	32,99 %
Southeastern Anatolia	2,34 %	2,46 %

East Anatolian	1,86 %	2,11 %
Mediterranean	7,44 %	6,97 %

After analyzing related past demand values, model will be firstly tested with all cities without grouping them so that the detailed pipeline system can be designed.

5.2.3. Cost Parameters

The related cost terms of Gunes (2013) will be used as the base case to determine the cost terms. Experimental runs are generated with Gunes (2013) values. After the experimental runs, cost terms of the mathematical model is determined as shown in Table 5.8.

Table 5.8. *Cost Parameters*

Parameters	Value
Storage Opening (cw)	2450 ($/month)
Pipe Establishment (co)	5980 ($/km-month)
Pipe Diameter (cp)	24 ($/month-inch-km)
Storage Holding (cht)	15 ($/MMscfd/month)
Purchase Cost (cbt)	70 ($/MMscfd)
Sale Cost (cst)	89 ($/MMscfd)

5.2.4. Model Results

PSNDP-B2 is used to solve the model. Node and arc information included in the model structure can be seen in Table 5.9.

Table 5.9. *Network Structure*

# of Demand Nodes	81
# of Storage Nodes	9
# of Supply Nodes	4
# of Arcs	253

The network system that is generated by the model can be seen in Figure 5.6. Six storage nodes are selected by the model. Three of the opened storage nodes are around Marmara region which has the highest demand with respect to other regions. The model choses to open storages around the biggest cities of Turkey such as Ankara, İzmir and Adana. No storage node is opened around east and west regions of Turkey. The model is solved within six iterations. The CPU time is 9420 seconds which is nearly two and half hour.

There exist similarities between the generated network configuration and the real Turkish pipeline system. Natural gas imported from east part of Turkey joins together at Erzurum in the current pipeline network. Similarly, the network configuration that is obtained by the model also is connected to each other around Erzurum. Natural gas route that is imported from Samsun yields the similar route that visit Amasya, Çorum, Kırıkkale and Ankara. The model creates the west pipeline system route by combining Denizli, Aydın, İzmir and Balıkesir. Indeed, west pipeline system of Turkey follows exactly the same route. Currently, Turkey has 2 storage facilities, which are Silivri and Tuz Gölü. The model choses to open existing storage facilities and four new storage facilities. There does not exist any pipeline connection between Eskişehir and Ankara in the model output. Instead of that pipeline arc, the model generates pipeline

arcs between Zonguldak and Düzce which is actually the planned pipeline extension project in the government agenda.

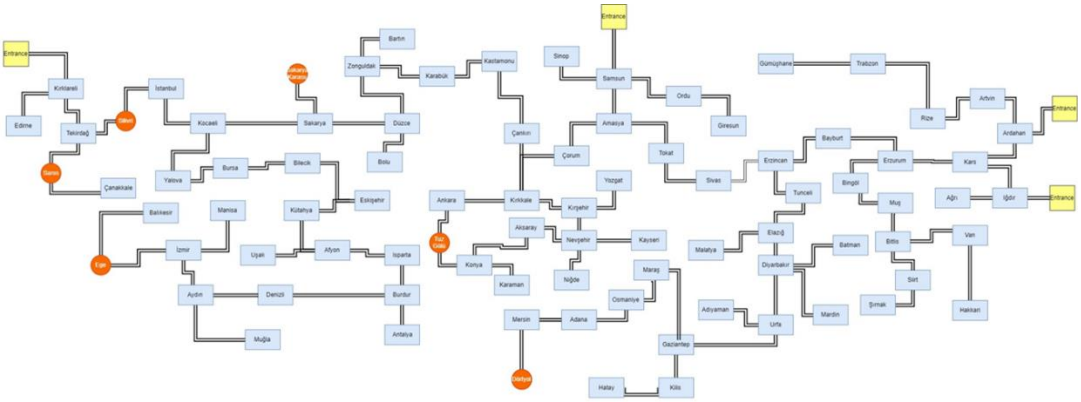

Figure 5.6. Model Solution Network

5.3. Stochastic Turkish Pipeline Case Study

The Stochastic Pipeline and Storage Network Design model is tested with the stochastic version of the Turkish case.

5.3.1. The Structure of Network

The network is created with 30 nodes. Highest 25 cities in terms of demand values are selected. Remaining cities are grouped into 5 nodes based on both proximity and their demand values. The locations of theses 5 nodes are found by calculating the averages of grouped city center locations. The network arcs that are included in the model are the ones between neighbor cities in the Turkish administrative map. Selected highest 25 cities are marked with red in the administrative map of Turkey in Figure 5.7. Grouped city center locations are marked with yellow color.

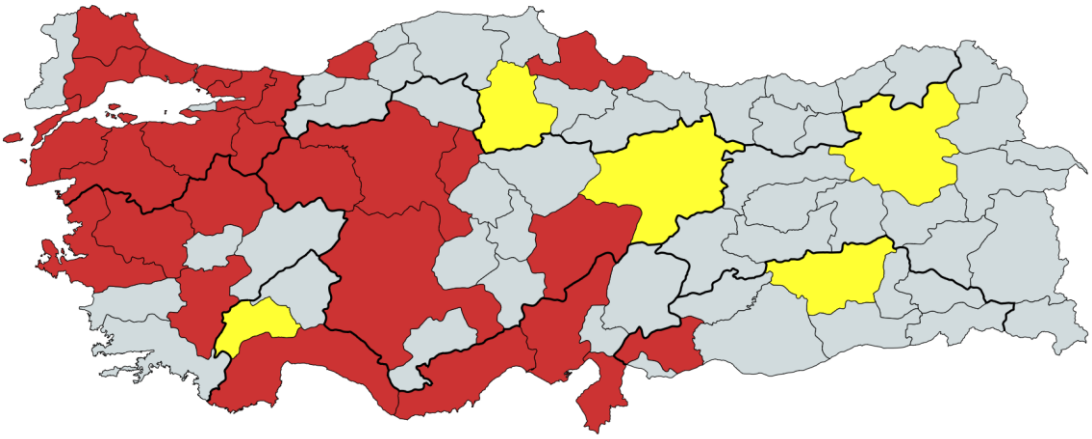

Figure 5.7. Selected Turkish cities

5.3.2. The Supply and Demand Values

Demand values are generated based on 2017. The demands of grouped nodes are equal to total demand of cities in the groups. The grouped cities are generally located in the east of Turkey so that their demand values are very low with respect to Ankara or İstanbul. It is stated in Table 5.7 that total demand percent is equal to 2.34% for Southeastern Anatolia cities and 1.86% for East Anatolian cites. Therefore, total demand value of grouped cities will not be so high to generate extreme values. Detailed information related nodes demand can be seen in below Table 5.10.

Table 5.10. *Model Parameters*

	Number of Nodes	Total demand MSm3	Percent
All Turkish cities	81	52498.87	-
Highest 25 cities	25	47182.83	89.87 %
Grouped cities	56	5316.87	10.13%

Detailed network elements can be seen in Table 5.11.

Table 5.11. *Network structure*

# of Demand Nodes	30
# of Storage Nodes	6
# of Supply Nodes	3
# of Arcs	135

5.3.3. Model Parameters

After generating deterministic model, the results of model yield pipeline system similar to current Turkish pipeline system. Therefore, cost terms of the deterministic model is used in this part.

5.3.4. Scenario Generation

The main aim of stochastic Turkish pipeline case study is to determine results of 'What-If-Analysis'. Similar procedure that is applied in randomly generated stochastic case study is applied. Node set is divided into three equal regions based on their proximity. There levels of demand values such as low, medium and high are applied in the analysis. The low demand values are selected from demand values in 2017 for high and low season. Medium and high demand values are generated by multiplying scenario factor with low demand value for both season. The provability of each demand levels is selected as equal to each other. Detailed conversion factors can be seen in Table 5.12. For each case, 27 scenarios are generated.

Table 5.12. *Scenario Factors*

	Scenario Factor Medium	Scenario Factor High
Case 1	1.2	2
Case 2	1.5	3
Case 3	2	6

In addition, the effect of probabilities of each scenario is analyzed. For case 1, the effect of increasing the probabilities of high demand values are analyzed. Applied probability values can be seen in Table 5.13.

Table 5.13. *Probabilities of Demand Values*

	Probability of Low	Probability of Medium	Probability of High
Case 4	1/6	1/6	2/3
Case 5	1/12	1/4	2/3

Analysis are divided into 5 main case stated as above. Currently, Turkey import most of its natural gas from Malkoçlar, Durusu and Gürbulak entrance points so that only three entrance points are included in the analysis. Moreover, yearly import amounts can be ranked from highest to lowest as Malkoçlar, Durusu and Gürbulak. These entrance points are labeled as Port1, Port 2 and Port 3. Malkoçlar entrance point which is located at Bulgarian borderline is used to transfer Russian gas. Durusu gate also carries Russian gas to Samsun. Natural gas of Iran is transferred to Turkey through Gürbulak gate.

5.3.5. Model Results

Stochastic model results can be seen in Table 5.14. In contrast to randomly generated stochastic case study, all problems are solved to optimality in stochastic Turkish pipeline study. The case studies are expected to easily solve due to network structure which consists of few high demand points around entrance nodes. Generated network structure maps are available in the Appendix-C.

Moreover, network structure of pipeline consists of less number of arcs. Network structure of case 1 is generated with cost of 735990.343 $. This category corresponds to equal probability and lowest scenario factor case. Opened storages are the ones around main connections points such as storages that are connected to more nodes rather than just high demand points. Opened storages are Ege, Tuz Gölü and Sakarya

Karasu. Opened storage are connecting at least two nodes so that stored gas distributed all network in case of decrement in supply.

Case 2 correspond the equal probability and medium level scenario factor. The generated network cost is equals to 754628.642 $. Increment in cost of network is expected because the network flow, storage amounts and pipe size increase. Model again chose to open Ege,Tuz Gölü and Sakarya Karasu storage nodes. Compared to Case 1, stored natural gas amount are higher at early times steps so that nodes are supplied from storage for longer time periods. The model time periods start with low demand season so that higher scenario factor can be the reason of higher storage amounts at early time steps. The effect of scenario factor is observed more after 5th time periods. The selected nodes are again connected more than one nodes. However, Sakarya Karasu storage connected to İstanbul at this time. Half of the stored gas is provided to İstanbul at Sakarya Karasu storage.

Equal probability and high level scenario factor is applied at Case 3. Generated network cost is equals 807754.039 $. Selected storage nodes are Silivri, Tuz Gölü, Ege and Dörtyol. It is first time that Dörtyol storage node is opened. The nodes are directly connected to high demand cities around them. For example, Silivri is connected to İstanbul. The stored gas is directly used to meet the demand at these nodes. However, only Dörtyol storage provide natural gas nodes around. In terms of network characteristic, entrance node two is directly connected to Ankara. However, this entrance node was connected to grouped node in Case 1 and Case 2. Moreover, the nodes around entrance two meet their demand from natural gas flow provide by entrance one. Apart from these, there isn’t too much difference between network structures of first three case. This observation is actually expected due the natural structure of network.

Probabilities of low and medium levels are equal to each other in Case 4. In addition, probability of high demand is bigger with respect to other two probabilities. Tuz Gölü, Silivri and Ege storage nodes are opened. Silivri and Tuz Gölü is directly provide flow

to Ankara and İstanbul. However, Ege storage is connected to İzmir and Balıkesir. Stored gas in the Ege storage is provided by all entrance nodes. In first three case study, stored gas only comes from the closest entrance points. Network structure is similar to Case 3.

In Case 5, probabilities of demand levels are increasing toward high demand values. Same storage nodes with Case 4 are opened. Flow from these nodes are mostly consumed by high demand cities around them. Pipeline structure is changed so that the flows are directly provided the high demand cities so that these nodes treated as hub location to distribute to gas. Moreover, lowest storage amounts are reached in this part. Main reason is the effect of probability level of high in the objective function.

There doesn't exist too many differences related with network characteristics of grouped cities because these nodes are located around the entrance points. The pipelines firstly visit these nodes.

Table 5.14. *Model Results*

Set No	CPU(s)	#of iteration	Gap(%)	Opt.	#of storage unit
Case 1	9141	4	0	735990.343	3
Case 2	9586	4	0	754628.642	3
Case 3	9689	5	0	807754.039	4
Case 4	9361	4	0	881607.531	3
Case 5	9578	5	0	882181.662	3

After solving the deterministic and stochastic versions of the Turkish pipeline case study, it should be needed to emphasize the results of the models. Both models choose to open storage units around the cities that has high population or industrial areas. Tuz

gölü and Silivri storages are selected in all the case studies. These storages are located around the high demand nodes. It is observed that the deterministic model opens four more storages nodes such as Adana, İzmir, Çanakkale and Sakarya, Moreover, these storages are generally located around the west of Turkey. These storages provide natural gases to cities around them not just for the one high demand city. It is also important that Silivri and Sakarya storages are also used to provide gas to İstanbul at the high demand terms.

In the stochastic mathematical model, selected storage nodes changes with respect to probabilities of scenarios. For the equal probability and lowest scenario factor case, the model opens Tuz gölü, Sakarya and Ege storages. These are also connected to cities around them so that they provide natural gas to the area around them. However, the selected storage nodes change to Silivri, Tuz gölü, Ege and Adana in the third case. It is observed that storage units start to connect directly to high demand cities and provide natural gas to directly to them such as İstanbul, Ankara and İzmir in case of high scenario factor and for high probability values. For the fourth and the fifth cases, selected storages show the similar characteristic. Flow from storage nodes provide gas to high demand cities. Same storage nodes are selected in these cases.

5.4. Discussion of Turkish Pipeline System

After conducting case studies, it is required to presents concluding remarks related with the actual Turkish pipeline system. The mathematical model selects the storage nodes around populated areas such as Ankara, İzmir, Adana and İstanbul. In addition, the opened storages are directly connected to these nodes. In case of high demand scenario, there exists two opened storage nodes in Marmara region. One of them store the natural gas that comes from the Bulgarian border. The other one is connected to the natural gas source from the Samsun gate.

In terms of pipeline arcs, there exist pipeline structure between Düzce and Zonguldak so that natural gas from Samsun can be pumped directly to Istanbul by following this pipeline. One of the important differences between the current system and the model

outcome is that there is no pipeline between Ankara and Istanbul. Bulgarian gate provides natural gas to cities in the west of Eskisehir. In the same way, Samsun gate provide natural gas to cities around Ankara. However, there exist huge similarities between the Turkish pipeline system and the model output. The natural gas pipelines at the east of Turkey are connected at Erzurum and pipeline structure in the west of Turkey is similar to the model output. Designing a pipeline system is a costly project so that at the 1980s Turkish cities is incorporated into the system one by one but the mathematical model is solved with all the Turkish cities. Therefore, differences between the actual system and the model output is expected.

CHAPTER 6

CONCLUSION

In this study, deterministic and two-stage stochastic mathematical models are developed for the pipeline and storage network design problem.

In the deterministic mathematical model, three different problems are tested. Deterministic mathematical models are solved with three different versions of Benders decomposition. In Benders decomposition procedures, the generation of sub-problems are different. These versions are Benders decomposition with extreme rays (PSNDP-B1), Benders decomposition with extreme points (PSNDP-B2) and Benders decomposition with maximum feasible subsystems cut generation (PSNDP-B3). These three methods produce different cuts that are appended to the master problem. The solution performances of these methods are compared. For the large sized problem, it becomes impractical to solve single stage model. The solution performances are reported with CPU time, number of Benders iterations, optimality gaps and number of optimal solutions found in each replications. All three versions of solution methods outperform single stage solution methods in terms of all performance measures. PSNDP-B2 solves all problems with a less number of iterations and CPU. Therefore, PSNDP-B2 is used in both two stage stochastic model and Turkish pipeline case study.

The mathematical model is generated for two stage stochastic model that optimize the expected profit. Expected profit is generated from sale of natural gas to end user minus purchasing and storage costs. In the two stage stochastic network design problem, the first stage model determines pipeline arcs and storage opening decisions. In the second stage, flow amount between nodes and storage decisions are made after realization of scenarios. In stochastic mathematical model, the model is tested with one problem

class with 30 nodes because solution times reach its limit with even 30 demand nodes, 2 storage nodes, 3 ports and 120 arcs. Nodes are divided into 3 main regions based on their closeness. 3 different demand levels such as low, medium and high are used to generate 27 scenarios. The number of iterations is higher with respect to the deterministic case. Only 12 problem sets are used with stochastic case due to time limits.

The mathematical model is used to design the Turkish pipeline network system. Demand values of 2017 are used. It is important that generated network system is fully connected so that the interruptions in one port will be solved by natural gas supplies from other ports. Generated network structure is similar to the current pipeline system in Turkey. Model chooses to open more store units around the Marmara region of Turkey which includes the main demand points.

Stochastic version of Turkish pipeline system is also modeled. The main aim is to determine the effect of demand values on the system. Scenario factors and different probability levels are applied in the models. Generated problems are divided into 5 main categories. Model results highlight the importance of storage facilities. In case of high demand conditions, storage nodes start to meet the natural gas demand of big cities directly. In addition, increment in the demand probabilities result in the network structure that storage nodes connected only the high demand cities.

Furthermore, there exist several future research extensions related with this study. One extension is to include compressor stations in the network design. By including compressor stations, the network operation cost in terms of fuel used by compressor can be included in the objective function. The compressor station also provides to yield network structure with pressure limits. Moreover, multi stage stochastic programing can be applied.

REFERENCES

André, J., Auray, S., Brac, J., DeWolf, D., Maisonnier, G., Ould-Sidi, M., et al.(2013).Design and dimensioning of hydrogen transmission pipeline networks. European Journal of Operational Research,229(1),239–251.

Asian Infrastructure Investment Bank (2018), Project Report of Turkish Gas Storage Expansion Project 2018. Retrieved from https://www.aiib.org/en/projects/approved/2018/golu-gas-storage-expansion.html

Benders, J. F., (1962), Partitioning Procedures for Solving Mixed-Variables Programming Problems, Numerische Mathematik, Vol. 4,

Bhaskaran, S., Salzborn, F., (1979) Optimal design of gas pipeline networks, Journal of the Operations Research Society, 30,1047–1060.

Birge, J. R. and Louveaux, F. (2011), Introduction to Stochastic Programming, Springer, New York, NY.

Binato, S., Pereira, F., Granville, S., (2001). A new Benders decomposition approach to solve power transmission network design problems, IEEE Trans. Power Systems, 16, 235–240.

British Petroleum Company (2019). BP Energy Outlook 2019. Retrieved from https://www.bp.com/content/dam/bp/business-sites/en/global/corporate/pdfs/energy-economics/energy-outlook/bp-energy-outlook-2019.pdf

British Petroleum Company (2018). BP Energy Outlook 2018. Retrieved from https://www.bp.com/content/dam/bp/business-sites/en/global/corporate/pdfs/energy-economics/energy-outlook/bp-energy-outlook-2018.pdf

Brimberg, J., Hansen, P., Lin K.-W., Mladenovic, N., Breton M., (2003). An Oil Pipeline Design Problem Operations Research, 51,228–239.

Butler, J.C. Dyer, J.S. Optimizing Natural Gas Flows with Linear Programing and Scenarios J.,

Corte, A. , Sörensen, K. (2013). Optimization of gravity-fed water distribution network design: a critical review. European Journal of Operational Research, 228, 1–10.

Costa, M., (2005). A survey on benders decomposition applied to fixed-charge network design problems. Computer& Operation Research , 32, 1429-1450

Cakir, O., (2009). Benders decomposition applied to multi-commodity, multi-mode distribution planning. Expert Systems with Applications, 36(4), 8212–8217

Dantzig, G. B. (1955), Linear programming under uncertainty. Management Science 1(3-4), 197–206.

D'Addiio, N., Abeygunawardana, A., Forbes, M., Ledwich, G., Shafiei. M., (2017) Approach to large distribution network optimization using modern implementation of benders decomposition. Open Access Proc. J., 1, 2110–2114.

Demissie, A., (2015). Multi Objective Optimization for Natural Gas Pipeline Network Operation. Lamar University.

Edgar, T., Himmelblau D., (1978) Optimal Design of Gas Transmission Network. Society of Petroleum Engineers Journal, 18(2).

Egging, R., Gabriel, S., Holz, F., Zhuang, J., (2007). A complementarity model for the European natural gas market, DIW Discussion Papers, 732,

Egging, R., Gabriel, S., (2006) Examining market power in the European natural gas market. Energy Policy, 34, 2762–2778

Energy Sector Regulatory Authority (2017). Natural Gas Market Sectorial Report 2017 Retrieved from https://www.epdk.org.tr/Detay/Icerik/3-0-94/yillik-sektor-raporu

Energy Sector Regulatory Authority (2018), Natural Gas Market Sectorial Report 2018 Retrieved from https://www.epdk.org.tr/Detay/Icerik/3-0-94/yillik-sektor-raporu

Ergün, Y., (2013) Multi-layer network design problems in telecommunication. (Doctoral Dissertation) Middle East Technical University, Ankara, Turkey.

Folga, S M, (2007) Natural gas pipeline technology overview. Decision and Information Sciences. doi:10.2172/925391.

Gabriel, S., Kiet, S., Zhuang, J., (2005) A mixed complementary based equilibrium model of natural gas markets. Operation Research, 53(5), ii-889

Gavish, B., (1983), Formulations and Algorithms for the Capacitated Minimal Directed Tree Problem. Journal of the Association for Computing Machinery, 30(1), 118–132.

Gendron, B., Scutellà, M., Garroppo, R., Nencioni, G., & Tavanti, L., (2014). A branch-and-Benders-cut method for nonlinear power design in green wireless local area networks. In Publication CIRRELT, 42.

Gunes, F., "Optimal design of a gas transmission network: A case study of the Turkish natural gas pipeline network system" (2013). Graduate Theses and Dissertations. 13294S.

Herran-Gonzalez A., Cruz J., De B., Risco-Martin, J.L., (2009). Modeling and simulation of a gas distribution pipeline network. Applied Mathematical Modelling, 33, 1584-1600.

International Energy Outlook (2019). World Energy Outlook 2019, Retrieved from: https://www.iea.org/reports/world-energy-outlook-2019

King AJ, Wallace SW (2012) Modeling with Stochastic Programming (Springer, New York)

Kleywegt, A.J., Shapiro, A. Homem-De-Mello, T. The sample average approximation method for stochastic discrete optimization, SIAM Journal of Optimization 12 (2001) 479–502.

Lee, C., Lee, K., Park, S., (2013) Benders decomposition approach for the robust network design problem with flow bifurcations. Networks 62(1), 1–16

Listes O (2007) A generic stochastic model for supply-and-return network design. Comput. Oper. Res. 34:417–442

Lise W., Hobbs B.F., (2008) Future evolution of the liberalized European gas market: Simulation results with a dynamic model. Energy, 33,989-1004

Mahey P, Benchakroun A, Boyer F., (2001) Capacity and flow assignment of data networks by generalized Benders decomposition. Journal of Global Optimization, 20,173–93

Marin, A., Jaramillo, P., (2009) Urban rapid transit network design: accelerated Benders decomposition. Annual Operation Research, 169,35-53

Mercado, R., Sanchez, C., (2015) Optimziation Problems in Natural Gas Transportation Systems: A state of- the-art review. Applied Energy, 147,536-555

Miranda, G., Luna, H., Camargo, R., Pinto, R, (2011). Tree network design avoiding congestion. Appl. Math. Modell. 35, 4175–4188

Natural Gas Market in Turkey. (n.d.). Retrieved from http://www.gazbir.org.tr/en/natural-gas-market-in-turkey/72.

Oliveira, G., Costa, A., Binato S., (1995). Large scale transmission network planning using optimization and heuristic techniques, IEEE Trans. Power Syst., 10, 1828–1834,

Randazzo, D.,, Luna L., Mahey P., (2001) Benders decomposition for local access network design with two technologies. Discrete Mathematics and Theoretical Computer Science, 4 235–246

Republic of Turkey Ministry of Energy and Natural Resources - Natural Gas Pipelines and Projects. (n.d.). Retrieved from https://enerji.gov.tr/en-US/Pages/Natural-Gas-Pipelines-and-Projects.

Romo, F., Tomasgard, A., Hellemo, L., Fodstad, M., Eidesen B.H., Pedersen, B.,(2009) Optimizing the Norwegian Natural Gas Production and Transport. 39(1):46-56

Rothfarb, B., Frank, H., Rosenbaum, D. M., Steiglitz, K., & Kleitman, D. J., (1970). Optimal design of offshore natural-gas pipeline systems. Operations Research, 18, 992–1020.

Rothfarb, B., Goldstein, M., , The One- Terminal TELPAK Problem, Operation Research. 1971; 19(1)

Saharidis, G., Ierapetritou, M., (2010) Improving benders decomposition using maximum feasible subsystem (MFS) cut generation strategy. Computers & Chemical Engineering, 24(8),1237-1245

Santoso, T., Ahmed, S., Goetschalckx, M., Shapiro, A., 2005. A stochastic programming approach for supply chain network design under uncertainty. European Journal of Operational Research 167,96–115.

Shiono, N., Suzuki, H., (2016) Optimal pipe-sizing problem of tree-shaped gas distribution networks. European Journal of Operational Research 252, 550–560

G., Supat K., Jifang Zhuang, (2005) A Mixed Complementarity-Based Equilibrium Model of Natural Gas Markets. Operations Research 53(5):799-818.

Tabkhi F., Pibouleau L., Azzaro-Pantel C. and Domenech S., (2009) Total Cost Minimization of a High-Pressure Natural Gas Network. Journal of Energy Resources Technology. 131, 3002-1

The Oxford Institute for Energy Studies (2018). Gas Supply Changes in Turkey, Retrieved from: https://www.oxfordenergy.org/wpcms/wp-content/uploads/2018/01/Gas-Supply-Changes-in-Turkey-Insight-24.pdf

Tometzki T., Engell, S., (2011) Systematic Initialization Techniques from Hybrid Evolutionary Algorithms for Solving Two Stage Stochastic Mixed Integer Programs, IEEE Transactions on Evolutionary Computation, 15(2),196-214

Turkey and the new energy politics of the Black Sea region. (n.d.). Retrieved from https://www.blackseanews.net/en/read/5757/.

Üster, H. and Hwang, S. O. (2016), 'Closed-loop supply chain network design under demand and return uncertainty', Transportation Science forthcoming.

Wheatley, D., Gzara, F., Jewkes, E., (2015). Logic-based Benders decomposition for an inventory-location problem with service constraints. Omega, 55, 10–23.

Wang, X., Porcu, M., (2008). The Potential of Compressed Natural Gas Transport in Asia. International Petroleum Technology Conference. doi:10.2523/IPTC-12078-MS

Wang, X., Porcu, M., (2008). The Economics of Compressed Natural Gas Sea Transport. SPE Russian Oil and Gas Technical Conference and Exhibition

Woldeyohannes A., Majid M., (2011) Simulation model for natural gas transmission pipeline network system Simulation Modelling Practice and Theory, 13(1),196-212

Wu, P., Hartman, J., Wilsom, G., (2003) A demand-shifting feasibility algorithm for Benders decomposition. European Journal of Operational Research, 148, 570–583

Zhuang J., (2005), A Stochastic equilibrium model for the north American natural gas market.

Zhang, H., Liang, Y., Ma, J., Qian, C., Yian, X., (2017) An MILP method for optimal offshore oilfield gathering system, Ocean Engineering, 141,25-34

APPENDICES

A. APPENDIX –A DETERMINISTIC MODEL INDIVIDUAL RESULTS

Table A.1. *Deterministic Model Individual Results*

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-1-1	10800	3.15	n32	130343.599	222	2	0	n31	127395.779
1-1-2	10800	3.49	n32	125094.790	4560	4	0	n31	123086.661
1-1-3	10800	4.56	n32	130872.029	820	2	0	n31	126755.134
1-1-4	10800	4.69	n32	132168.718	1600	2	0	n31	128633.064
1-1-5	10800	3.39	n32	127489.817	625	2	0	n31	124366.915
Average	10800	3.856		129193.791	1565.400	2.4	0		126047.511

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-1-1	229	4	0	n31	127395.779	1650	3	0	n31	127395.779
1-1-2	1928	4	0	n31	123086.661	1593	3	0	n31	123086.661
1-1-3	757	3	0	n31	126755.134	1819	3	0	n31	126755.134
1-1-4	1544	3	0	n31	128633.064	1683	3	0	n31	128633.064
1-1-5	592	3	0	n31	124366.915	1704	3	0	n31	124366.915
Average	1010	3.4	0		126047.511	1689.8	3	0		126047.511

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-2-1	10800	5.14	n32	131437.417	1810	4	0	n31	126884.679
1-2-2	10800	5.21	n32	129643.261	1491	4	0	n31	122647.712
1-2-3	10800	4.79	n31	127888.407	781	4	0	n31	124949.501
1-2-4	10800	5.79	n32	132050.352	2434	4	0	n31	128437.369
1-2-5	10800	4.44	n32	127232.77	1877	4	0	n31	123998.106
Average	10800	5.074		129650.4414	1678.6	4	0		125383.4734

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-2-1	878	3	0	n32	126884.679	1779	5	0	n32	126884.679
1-2-2	2432	4	0	n32	122647.712	1644	5	0	n32	122647.712
1-2-3	965	4	0	n32	124949.501	1775	6	0	n32	124949.501
1-2-4	1205	3	0	n32	128437.369	1538	5	0	n32	128437.369
1-2-5	952	4	0	n32	123998.106	1980	5	0	n32	123998.106
Average	1286.4	3.6	0		125383.4734	1743.2	5.2	0		125383.4734

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-3-1	10800	5.14	n32	130266.597	2239	3	0	n32	125845.615
1-3-2	10800	5.21	n32	125643.621	1903	3	0	n32	122597.98
1-3-3	10800	4.79	n31	127968.407	2195	3	0	n32	124680.518
1-3-4	10800	5.79	n32	132050.952	1992	3	0	n32	128302.416
1-3-5	10800	4.44	n32	127232.77	2036	3	0	n32	125855.827
Average	10800	5.074		128632.4694	2073	3	0	n32	125456.4712

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-3-1	909	4	0	n32	125845.615	2383	5	0	n32	125845.615
1-3-2	2721	3	0	n32	122597.98	2086	5	0	n32	122597.98
1-3-3	1003	3	0	n32	124680.518	2124	4	0	n32	124680.518
1-3-4	1408	3	0	n32	128302.416	2237	4	0	n31	128302.416
1-3-5	883	3	0	n32	125855.827	2208	5	0	n31	125855.827
Average	1384.8	3.2	0		125456.4712	2207.6	4.6	0		125456.4712

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-4-1	10800	4.17	n33	134111.11	1695	4	0	n31	131903.246
1-4-2	10800	3.95	n33	127016.741	1653	4	0	n31	125168.924
1-4-3	10800	4.19	n33	131705	1924	4	0	n31	128549.065
1-4-4	10800	5.67	n33	131056.075	1402	4	0	n31	127422.894
1-4-5	10800	5.42	n33	132424.086	1350	4	0	n31	128774.032
Average	10800	4.68		131262.6024	1604.8	4	0		128363.6322

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-4-1	844	4	0	n31	131903.246	1748	5	0	n31	131903.246
1-4-2	940	5	0	n31	125168.924	1938	6	0	n33	125168.924
1-4-3	1303	4	0	n31	128549.065	1831	5	0	n34	128549.065
1-4-4	1117	4	0	n34	127422.894	2008	5	0	n34	127422.894
1-4-5	1382	5	0	n32	128774.032	2092	5	0	n32	128774.032
Average	1117.2	4.4	0		128363.6322	1923.4	5.2	0		128363.6322

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-5-1	10800	5.38	n33	134524.656	1278	4	0	n31	131394.819
1-5-2	10800	5.54	n33	127370.049	2645	4	0	n31	124985.706
1-5-3	10800	6.76	n33	132342.233	2599	4	0	n31	127812.733
1-5-4	10800	10.06	n33	134830.126	1413	5	0	n31	126822.604
1-5-5	10800	8.94	n33	135811.372	1303	4	0	n31	128351.514
Average	10800	7.336		132975.6872	1847.6	4.2	0		127873.4752

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-5-1	1390	3	0	n31	131394.819	2254	5	0	n31	131394.819
1-5-2	1084	5	0	n31	124985.706	2525	5	0	n31	124985.706
1-5-3	1305	4	0	n31	127812.733	2138	5	0	n34	127812.733
1-5-4	1230	4	0	n31	126822.604	2107	5	0	n33	126822.604
1-5-5	1255	4	0	n31	128351.514	2287	5	0	n31	128351.514
Average	1252.8	4	0		127873.4752	2262.2	5	0		127873.4752

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-6-1	10800	10.19	n33	141076.295	1390	5	0	n31	131291.728
1-6-2	10800	10.29	n33	133264.796	2458	5	0	n31	125286.221
1-6-3	10800	8.32	n33	133563.831	2911	5	0	n31	127511.404
1-6-4	10800	16.19	n33-n34	143829.719	1820	5	0	n31	125225.886
1-6-5	10800	11.82	n33	139133.985	2086	5	0	n31	128136.553
Average	10800	11.362		138173.7252	2133	5	0		127490.3584

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-6-1	1207	3	0	n31	131291.728	2202	5	0	n33	131291.728
1-6-2	1303	3	0	n31	125286.221	2248	5	0	n31	125286.221
1-6-3	1408	3	0	n31	127511.404	2459	5	0	n34	127511.404
1-6-4	1294	4	0	n31	125225.886	2315	5	0	n34	125225.886
1-6-5	1514	4	0	n31	128136.553	2502	5	0	n32	128136.553
Average	1345.2	3.4	0		127490.3584	2345.2	5	0		127490.3584

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-7-1	10800	5.38	n31-n32	133530.353	1544	5	0	n31 n32	131250.767
1-7-2	10800	4.56	n31-n32	128210.5	1827	6	0	n31 n32	126816.409
1-7-3	10800	4.57	n31-n32	125641.102	1897	5	0	n31 n32	123919.761
1-7-4	10800	5.63	n31-n32	133609.787	1441	5	0	n31, n32	130408.716
1-7-5	10800	5.38	n31-n32	130439.556	1514	5	0	n31, n32	127317.645
Average	10800	5.104		130286.2596	1644.6	5.2	0		127942.6596

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-7-1	1089	4	0	n31 n32	131250.767	2541	6	0	n31 n32	131250.767
1-7-2	1293	4	0	n31 n32	126816.409	2443	6	0	n31 n32	126816.409
1-7-3	1287	4	0	n31 n32	123919.761	2426	6	0	n31 n32	123919.761
1-7-4	948	4	0	n31, n32	130408.716	2817	6	0	n31 n32	130408.716
1-7-5	1169	4	0	n31, n32	127317.645	2049	6	0	n31, n32	127317.645
Average	1157.2	4	0		127942.6596	2455.2	6	0		127942.6596

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-8-1	10800	6.71	n31 n32	132867.696	2160	5	0	n31 n32	128839.722
1-8-2	10800	6.7	n31 n32	129713.466	1654	5	0	n31 n32	126392.362
1-8-3	10800	7.91	n31,n32	127504.077	2039	5	0	n31 n32	123297.26
1-8-4	10800	10.25	n31 n32	138663.939	1998	5	0	n31 n32	130298.278
1-8-5	10800	7.39	n31 n32	132597.603	2102	5	0	n31 n32	127359.235
Average	10800	7.792		132269.3562	1990.6	5	0		127237.3714

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-8-1	1481	5	0	n31 n32	128839.722	2335	6	0	n31 n32	128839.722
1-8-2	1583	5	0	n31 n32	126392.362	2039	6	0	n31 n32	126392.362
1-8-3	936	5	0	n31 n32	123297.26	2777	5	0	n31 n32	123297.26
1-8-4	1305	5	0	n31 n32	130298.278	2728	6	0	n31 n32	130298.278
1-8-5	1334	5	0	n31 n32	127359.235	2450	6	0	n31 n32	127359.235
Average	1327.8	5	0		127237.3714	2465.8	5.8	0		127237.3714

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-9-1	10800	13.73	n31, n32	141904.211	1435	5	0	n31, n32	128473.612
1-9-2	10800	12.41	n31, n32	137139.565	924	6	0	n31, n32	126281.152
1-9-3	10800	10.68	n31 n32	129365.981	5678	6	0	n31, n32	122917.966
1-9-4	10800	7.57	n31, n32	134174.435	2181	5	0	n31, n32	130298.278
1-9-5	10800	11.88	n31, n32	138712.744	997	5	0	n31 n32	127359.235
Average	10800	11.254		136259.3872	2243	5.4	0		127066.0486

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-9-1	2335	5	0	n31, n32	128473.612	2497	5	0	n31 n32	128473.612
1-9-2	2039	5	0	n31, n32	126281.152	2273	6	0	n31 n32	126281.152
1-9-3	2777	6	0	n31, n32	122917.966	2490	6	0	n31 n32	122917.966
1-9-4	2728	6	0	n31, n32	130298.278	2987	6	0	n31 n32	130298.278
1-9-5	2450	6	0	n31 n32	127359.235	2786	6	0	n31 n32	127359.235
Average	2465.8	5.6	0		127066.0486	2606.6	5.8	0		127066.0486

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-10-1	10800	5.97	n33,n34	131032.044	1971	5	0	n31 n32	129310.375
1-10-2	10800	7.03	n32, n33	128031.125	1963	5	0	n31 n32	124369.201
1-10-3	10800	6.73	n31, n34	135892.462	1623	5	0	n31 n32	133498.469
1-10-4	10800	6.45	n32, n33	128047.039	1897	5	0	n31 n32	125931.947
1-10-5	10800	8.06	n32, n33	135192.371	1774	5	0	n31 n32	131376.86
Average	10800	6.848		131639.0082	1845.6	5	0		128897.3704

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-10-1	1259	5	0	n31 n32	129310.375	2414	5	0	n31 n32	129310.375
1-10-2	1175	5	0	n31 n32	124369.201	2455	5	0	n31 n32	124369.201
1-10-3	1560	5	0	n31 n32	133498.469	2617	5	0	n31 n32	133498.469
1-10-4	1398	5	0	n31 n32	125931.947	2550	5	0	n31 n32	125931.947
1-10-5	1201	5	0	n31 n32	131376.86	2449	5	0	n31 n32	131376.86
Average	1318.6	5	0		128897.3704	2497	5	0		128897.3704

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-11-1	10800	8.38	n33, n34	132683.612	1480	5	0	n31 n32	128912.481
1-11-2	10800	10.15	n33, n34	130927.879	1503	5	0	n31 n32	123903.322
1-11-3	10800	6.99	n31, n34	133688.651	1751	5	0	n31 n32	130696.223
1-11-4	10800	11.83	n32, n34	133914.598	1432	5	0	n31 n32	125104.909
1-11-5	10800	13.81	n31, n33	142489.824	1636	5	0	n31 n32	129835.704
Average	10800	10.232		134740.9128	1560.4	5	0		127690.5278

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-11-1	2052	5	0	n31 n32	128912.481	2639	5	0	n32 n34	128912.481
1-11-2	2024	5	0	n31 n32	123903.322	2649	5	0	n32 n33	123903.322
1-11-3	2524	5	0	n31 n32	130696.223	2087	5	0	n33 n34	130696.223
1-11-4	2275	5	0	n31 n32	125104.909	2455	5	0	n33 n34	125104.909
1-11-5	2186	5	0	n31 n32	129835.704	2724	5	0	n31 n32	129835.704
Average	2212.2	5	0		127690.5278	2510.8	5	0		127690.5278

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-12-1	10800	10.58	n32,n34	135134.595	1459	5	0	n31 n32	128214.6
1-12-2	10800	13.37	n32, n33	135378.975	1583	5	0	n31 n32	123562.735
1-12-3	10800	11.7	n33, n34	139345.116	1600	5	0	n31 n32	130088.477
1-12-4	10800	14.53	n32, n33	136783.776	1435	5	0	n31 n32	124538.667
1-12-5	10800	12.34	n32, n33	139813.477	1300	5	0	n31 n32	129431.72
Average	10800	12.504		137291.1878	1475.4	5	0		127167.2398

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
1-12-1	2687	5	0	n31 n32	128214.6	2577	6	0	n32 n34	128214.6
1-12-2	2428	5	0	n31 n32	123562.735	2740	5	0	n33 n34	123562.735
1-12-3	2253	5	0	n31 n32	130088.477	2936	5	0	n33 n34	130088.477
1-12-4	2152	5	0	n31 n32	124538.667	2386	5	0	n32 n34	124538.667
1-12-5	2024	5	0	n31 n32	129431.72	2881	5	0	n31 n34	129431.72
Average	2308.8	5	0		127167.2398	2704	5.2	0		127167.2398

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-1-1	10800	9.96	n42	171783.33	4111	3	0	n42	160144.421
4-1-2	10800	6.38	n41	164567.973	4196	3	0	n41	159823.113
4-1-3	10800	7.6	n42	168585.017	4319	3	0	n42	162140.207
4-1-4	10800	12.08	n42	179253.447	4231	3	0	n42	163348.579
4-1-5	10800	8.42	n42	168632.43	4196	3	0	n42	161437.684
Average	10800			170564.4394	4210.6	3	0		161378.8008

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-1-1	3995	3	0	n42	160144.421	4722	3	0	n42	160144.421
4-1-2	3968	3	0	n41	159823.113	4779	4	0	n41	159823.113
4-1-3	4019	3	0	n42	162140.207	4839	5	0	n42	162140.207
4-1-4	4017	3	0	n42	163348.579	4878	4	0	n42	163348.579
4-1-5	4001	3	0	n42	161437.684	4795	4	0	n42	161437.684
Average	4000	3	0		161378.8008	4802.6		0		161378.8008

	PSNDP				PSNDP-B1-dummy				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-2-1	10800	10.96	n42	165642.236	4356	3	0	n42	160529.225
4-2-2	10800	9.1	n41	174106.697	4381	4	0	n41	158411.58
4-2-3	10800	9.67	n42	170999.268	4588	6	0	n42	160870.586
4-2-4	10800	10.3	n42	182281.002	4333	4	0	n41	162994.102
4-2-5	10800	10.47	n41	176722.431	4433	3	0	n42	160200.842
Average	10800	10.1		173950.3268	4418.2	4	0		160601.267

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-2-1	4184	4	0	n42	160529.225	4803	4	0	n42	160529.225
4-2-2	4322	3	0	n41	158411.58	4903	4	0	n41	158411.58
4-2-3	4203	5	0	n42	160870.586	5086	6	0	n42	160870.586
4-2-4	4148	3	0	n41	162994.102	4935	3	0	n41	162994.102
4-2-5	4158	4	0	n42	160200.842	5018	3	0	n42	160200.842
Average	4203	3.8	0		160601.267	4949	4	0		160601.267

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-3-1	10800	10.24	n42	177770.854	4808	4	0	n42	159246.322
4-3-2	10800	13.8	n41	170219.818	4855	4	0	n41	157480.578
4-3-3	10800	11.35	n41	176546.894	4931	5	0	n42	160122.985
4-3-4	10800	10.35	n41 n42	181090.219	4825	3	0	n42	162763.101
4-3-5	10800	9.31	n42	182529.567	5084	4	0	n42	159815.809
Average	10800	11.01		177631.4704	4900.6	4	0		159885.759

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-3-1	4650	5	0	n42	159246.322	5076	4	0	n42	159246.322
4-3-2	4771	5	0	n41	157480.578	5188	4	0	n41	157480.578
4-3-3	4617	5	0	n42	160122.985	5169	4	0	n42	160122.985
4-3-4	4704	5	0	n42	162763.101	5040	4	0	n42	162763.101
4-3-5	4728	5	0	n42	159815.809	5153	4	0	n42	159815.809
Average	4694	5	0		159885.759	5125.2	4	0		159885.759

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-4-1	10800	8.62	n43	174951.368	4496	6	0	n43	170684.333
4-4-2	10800	7.78	n44	166691.898	4494	4	0	n42	161966.769
4-4-3	10800	8.56	n41	170177.998	4657	4	0	n44	163093.949
4-4-4	10800	11.14	n41	178254.787	4652	6	0	n41	191261.084
4-4-5	10800	8.4	n42	167944.59	4625	5	0	n43	164223.742
Average	10800	8.9		171604.1282	4584.8	5	0		170245.9754

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-4-1	4366	6	0	n43	170684.333	4701	6	0	n43	170684.333
4-4-2	4349	3	0	n42	161966.769	4681	3	0	n42	161966.769
4-4-3	4382	5	0	n44	163093.949	4712	5	0	n44	163093.949
4-4-4	4282	3	0	n41	191261.084	4744	4	0	n41	191261.084
4-4-5	4358	5	0	n43	164223.742	4735	5	0	n43	164223.742
Average	4347.4	4.4	0		170245.9754	4714.6	4.6	0		170245.9754

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-5-1	10800	11.53	n43	186279.7	4614	4	0	n43	169665.979
4-5-2	10800	11.18	n44	171267	4557	4	0	n44	161292.907
4-5-3	10800	10.2	n42	175254	4718	4	0	n43	159923.974
4-5-4	10800	13.24	n41 n42	184627.6	4585	4	0	n41	161945.143
4-5-5	10800	11.35	n41	175738.7	4560	4	0	n41	186482.071
Average	10800	11.5		178633.4	4606.8	4	0		167862.0148

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-5-1	4389	4	0	n43	169665.979	4969	4	0	n43	169665.979
4-5-2	4425	3	0	n44	161292.907	4897	4	0	n44	161292.907
4-5-3	4468	5	0	n43	159923.974	4924	4	0	n43	159923.974
4-5-4	4385	5	0	n41	161945.143	4868	4	0	n41	161945.143
4-5-5	4417	4	0	n41	186482.071	4874	4	0	n41	186482.071
Average	4416.8	4.2	0		167862.0148	4906.4	4	0		167862.0148

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-6-1	10800	11.53	n41	185953.668	5076	4	0	n43	168113.762
4-6-2	10800	11.68	n43	179912.494	5900	5	0	n41	160884.879
4-6-3	10800	10.6	n43	165642.726	4989	4	0	n43	158742.801
4-6-4	10800	15.84	n41	181788.075	4969	4	0	n41	162062.382
4-6-5	10800	14.35	n42	175482.009	4961	4	0	n41	161724.509
Average	10800	12.8		177755.7944	5179	4.2	0		162305.6666

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-6-1	4705	4	0	n43	168113.762	5485	4	0	n43	168113.762
4-6-2	4829	6	0	n41	160884.879	5256	4	0	n41	160884.879
4-6-3	4810	5	0	n43	158742.801	5220	4	0	n43	158742.801
4-6-4	4801	5	0	n41	162062.382	5361	4	0	n41	162062.382
4-6-5	4807	5	0	n41	161724.509	5076	4	0	n41	161724.509
Average	4790.4	5	0		162305.6666	5279.6	4	0		162305.6666

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-7-1	10800	9.3	n41 n42	174569.833	4933	4	0	n41 n42	159478.755
4-7-2	10800	7	n41 n42	177286.09	4844	4	0	n41 n42	159135.988
4-7-3	10800	6.62	n41 n42	166530.672	4796	4	0	n41 n42	159854.79
4-7-4	10800	6.18	n41 n42	178343.58	4921	4	0	n41 n42	164641.879
4-7-5	10800	5.4	n41 n42	175953	4990	4	0	n41 n42	159991.405
Average	10800	6.9		174536.635	4896.8	4	0	n41 n42	160620.5634

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-7-1	4772	6	0	n41 n42	159478.755	5115	4	0	n41 n42	159478.755
4-7-2	4766	5	0	n41 n42	159135.988	4980	4	0	n41 n42	159135.988
4-7-3	4860	6	0	n41 n42	159854.79	4919	4	0	n41 n42	159854.79
4-7-4	4798	4	0	n41 n42	164641.879	4942	4	0	n41 n42	164641.879
4-7-5	4716	7	0	n41 n42	159991.405	5155	5	0	n41 n42	159991.405
Average	4782.4	5.6	0		160620.5634	5022.2	4.2	0		160620.5634

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-8-1	10800	10.44	n41 n42	175905.993	5076	4	0	n41 n42	158884.528
4-8-2	10800	10.42	n41 n42	159349.038	5334	4	0	n41 n42	157274.582
4-8-3	10800	10.45	n41 n42	175961.984	5137	4	0	n41 n42	158710.208
4-8-4	10800	10.23	n41 n42	176842.493	5187	4	0	n41 n42	162895.733
4-8-5	10800	9.96	n41 n42	178332.196	5282	4	0	n41 n42	159630.676
Average	10800	10.3		173278.3408	5203.2	4	0	n41 n42	159479.1454

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-8-1	5461	4	0	n41 n42	158884.528	5560	4	0	n41 n42	158884.528
4-8-2	5467	4	0	n41 n42	157274.582	5585	4	0	n41 n42	157274.582
4-8-3	5523	5	0	n41 n42	158710.208	5614	4	0	n41 n42	158710.208
4-8-4	5396	4	0	n41 n42	162895.733	5634	4	0	n41 n42	162895.733
4-8-5	5506	5	0	n41 n42	159630.676	5821	5	0	n41 n42	159630.676
Average	5470.6	4.4	0		159479.1454	5642.8	4.2	0		159479.1454

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-9-1	10800	14.21	n41 n42	185707.460	5431	4	0	n41 n42	158658.851
4-9-2	10800	13.6	n41 n42	171974.215	5541	5	0	n41 n42	157158.825
4-9-3	10800	13.79	n41 n42	182279.755	5483	3	0	n41 n42	158071.793
4-9-4	10800	14.4	n41 n42	185249.685	5445	4	0	n41 n42	162405.485
4-9-5	10800	12.5	n41 n42	184946.402	5450	4	0	n41 n42	159288.871
Average	10800	13.7		182032	5470	4	0		159116.765

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-9-1	5925	5	0	n41 n42	158658.851	5907	4	0	n41 n42	158658.851
4-9-2	5900	5	0	n41 n42	157158.825	6209	5	0	n41 n42	157158.825
4-9-3	5911	4	0	n41 n42	158071.793	6053	4	0	n41 n42	158071.793
4-9-4	5973	5	0	n41 n42	162405.485	5962	4	0	n41 n42	162405.485
4-9-5	5859	4	0	n41 n42	159288.871	6114	4	0	n41 n42	159288.871
Average	5913.6	4.6	0		159116.765	6049	4.2	0		159116.765

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-10-1	10800	6.8	n43 n44	174587.2	5342	5	0	n41 n44	165941.880
4-10-2	10800	8.1	n43 n44	173793.4	5316	4	0	n43 n44	161685.752
4-10-3	10800	7.6	n43 n44	173949.6	5341	4	0	n43 n44	161575.411
4-10-4	10800	7.2	n43 n44	177622.5	5061	4	0	n41 n44	167829.270
4-10-5	10800	5.8	n43 n44	177344.7	5089	4	0	n41 n44	163129.593
Average	10800	7.1		175459.5	5229.8	4.2	0		164032.3812

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-10-1	4971	7	0	n41 n44	165941.880	5856	4	0	n41 n44	165941.880
4-10-2	4932	5	0	n43 n44	161685.752	5517	4	0	n43 n44	161685.752
4-10-3	4901	5	0	n43 n44	161575.411	5616	4	0	n43 n44	161575.411
4-10-4	4825	4	0	n41 n44	167829.270	5531	4	0	n41 n44	167829.270
4-10-5	4900	5	0	n41 n44	163129.593	5487	4	0	n41 n44	163129.593
Average	4905.8	5.2	0		164032.3812	5601.4	4	0		164032.3812

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-11-1	10800	10.04	n43 n44	178005.736	5811	4	0	n41 n44	164971.177
4-11-2	10800	11.69	n43 n44	176447.512	5850	4	0	n43 n44	161254.299
4-11-3	10800	11.94	n41 n42	171587.505	5637	4	0	n43 n44	158471.161
4-11-4	10800	10.91	n41 n44	185964.407	5873	4	0	n41 n44	164180.803
4-11-5	10800	9.91	n42 n43	181269.122	5709	4	0	n41 n44	162136.173
Average	10800	10.9		178654.8564	5776	4	0		162202.7226

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-11-1	5503	6	0	n41 n44	164971.177	5869	5	0	n41 n44	164971.177
4-11-2	5544	6	0	n43 n44	161254.299	5832	4	0	n43 n44	161254.299
4-11-3	5491	4	0	n43 n44	158471.161	6048	4	0	n43 n44	158471.161
4-11-4	5417	5	0	n41 n44	164180.803	6187	4	0	n41 n44	164180.803
4-11-5	5467	4	0	n41 n44	162136.173	5953	4	0	n41 n44	162136.173
Average	5484.4	5	0		162202.7226	5977.8	4.2	0		162202.7226

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-12-1	10800	14.26	n42 n43	177930.570	5865	4	0	n41 n44	164972.284
4-12-2	10800	14.23	n41 n42	182871.526	6188	6	0	n43 n44	160155.935
4-12-3	10800	13.8	n43 n44	185113.932	5912	5	0	n43 n44	158092.790
4-12-4	10800	15.51	n41 n44	184828.107	5970	4	0	n41 n44	163581.881
4-12-5	10800	14.7	n41 n42 n43	192663.025	5905	4	0	n41 n44	161598.069
Average	10800	14.5		184681.432	5968	4.6	0		161680.1918

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
4-12-1	5769	5	0	n41 n44	164972.284	5915	5	0	n41 n44	164972.284
4-12-2	5915	7	0	n43 n44	160155.935	6297	5	0	n43 n44	160155.935
4-12-3	5801	6	0	n43 n44	158092.790	6022	5	0	n43 n44	158092.790
4-12-4	5805	5	0	n41 n44	163581.881	6011	4	0	n41 n44	163581.881
4-12-5	5740	5	0	n41 n44	161598.069	5984	4	0	n41 n44	161598.069
Average	5806	5.6	0		161680.1918	6045.8	4.6	0		161680.1918

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-1-1	10800	5,49	n51	199951,979	5132	3	0	n51	196587,59
3-1-2	10800	6,05	n51	201338,207	5063	3	0	n51	197617,931
3-1-3	10800	6,34	n51	205033,801	5145	3	0	n51	199804,935
3-1-4	10800	8,68	n52	212461,485	5092	4	0	n51	202291,54
3-1-5	10800	5,62	n52	201793,422	4966	3	0	n51	197391,288
Average	10800	6,436		204115,7788	5079,6	3,2	0		198738,6568

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-1-1	4733	3	0	n51	196587,59	5715	4	0	n51	196587,59
3-1-2	4825	3	0	n51	197617,931	5568	4	0	n51	197617,931
3-1-3	4757	3	0	n51	199804,935	5676	4	0	n51	199804,935
3-1-4	4715	3	0	n51	202291,54	5665	4	0	n51	202291,54
3-1-5	4794	3	0	n51	197391,288	5644	4	0	n51	197391,288
Average	4764,8	3	0		198738,6568	5653,6	4	0		198738,6568

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-2-1	10800	6,92	n52	200046,068	5536	3	0	n52	196346,695
3-2-2	10800	8,99	n52	206173,814	5742	3	0	n52	196602,141
3-2-3	10800	9,48	n51	209798,713	5694	4	0	n52	199092,996
3-2-4	10800	9,32	n51	206147,66	5503	4	0	n51	195817,002
3-2-5	10800	11,05	n52	210299,858	5707	4	0	n51	195193,931
Average	10800	9,152		206493,2226	5636,4	3,6	0		196610,553

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-2-1	5290	4	0	n52	196346,695	6413	4	0	n52	196346,695
3-2-2	5373	4	0	n52	196602,141	6427	4	0	n52	196602,141
3-2-3	5404	3	0	n52	199092,996	6377	4	0	n52	199092,996
3-2-4	5324	3	0	n51	195817,002	6121	4	0	n51	195817,002
3-2-5	5358	3	0	n51	195193,931	6037	5	0	n51	195193,931
Average	5349,8	3,4	0		196610,553	6275	4,2	0		196610,553

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-3-1	10800	11,39	n51	210058,698	6186	4	0	n51	195838,093
3-3-2	10800	13,77	n52	216842,638	5940	4	0	n51	195911,936
3-3-3	10800	14,54	n52	221640,701	6035	4	0	n51	198622,876
3-3-4	10800	16,09	n52	221311,971	5971	4	0	n51	194774,524
3-3-5	10800	12,92	n52	213947,345	6171	4	0	n51	194320,405
Average	10800	13,742		216760,2706	6060,6	4	0		195893,5668

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-3-1	5842	4	0	n51	195838,093	6813	4	0	n51	195838,093
3-3-2	5873	4	0	n51	195911,936	6845	4	0	n51	195911,936
3-3-3	5859	4	0	n51	198622,876	6697	4	0	n51	198622,876
3-3-4	5805	4	0	n51	194774,524	6653	5	0	n51	194774,524
3-3-5	5801	4	0	n51	194320,405	6525	5	0	n51	194320,405
Average	5836	4	0		195893,5668	6706,6	4,4	0		195893,5668

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-4-1	10800	9,4	n53	199952,486	5281	4	0	n51	194439,204
3-4-2	10800	9,6	n53	207677,902	5080	4	0	n51	202668,667
3-4-3	10800	12	n53	215645,111	5007	4	0	n53	197508,065
3-4-4	10800	4,5	n54	214848,509	5104	4	0	n53	206110,825
3-4-5	10800	8,2	n51	211434,322	5186	4	0	n53	198573,959
Average	10800	8,525		209911,666	5131,6	4	0		199860,144

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-4-1	4833	4	0	n51	194439,204	5045	4	0	n51	194439,204
3-4-2	4907	4	0	n51	202668,667	5365	5	0	n51	202668,667
3-4-3	4857	4	0	n53	197508,065	5362	4	0	n53	197508,065
3-4-4	4860	4	0	n53	206110,825	5357	4	0	n53	206110,825
3-4-5	4969	4	0	n53	198573,959	5336	4	0	n53	198573,959
Average	4885,2	4	0		199860,144	5293	4,2	0		199860,144

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-5-1	10800	8,53	n53	200995,473	5740	4	0	n53	193112,081
3-5-2	10800	11,18	n53	215189,196	5588	4	0	n51	199998,768
3-5-3	10800	10,7	n53	209100,52	5594	4	0	n51	196360,412
3-5-4	10800	15,45	n53	226548,199	5601	4	0	n51	201286,985
3-5-5	10800	15,4	n51 n54	221154,623	5524	4	0	n51	196939,342
Average	10800	12,252		214597,6022	5609,4	4	0		197539,5176

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-5-1	5263	4	0	n53	193112,081	6012	4	0	n53	193112,081
3-5-2	5452	3	0	n51	199998,768	5866	5	0	n51	199998,768
3-5-3	5410	4	0	n51	196360,412	5705	5	0	n51	196360,412
3-5-4	5292	4	0	n51	201286,985	5789	4	0	n51	201286,985
3-5-5	5333	4	0	n51	196939,342	5885	4	0	n51	196939,342
Average	5350	3,8	0		197539,5176	5851,4	4,4	0		197539,5176

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-6-1	10800	11,89	n53	208341,484	6232	4	0	n51	192951,629
3-6-2	10800	8,81	n53	209516,063	6024	4	0	n51	200245,989
3-6-3	10800	14,98	n53	219209,125	6143	4	0	n51	195392,721
3-6-4	10800	15,99	n54	226709,797	6017	5	0	n51	199978,84
3-6-5	10800	14,32	n51	217699,189	6062	4	0	n51	195184,897
Average	10800	13,198		216295,1316	6095,6	4,2	0		196750,8152

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-6-1	5834	4	0	n51	192951,629	6179	4	0	n51	192951,629
3-6-2	5742	4	0	n51	200245,989	6087	4	0	n51	200245,989
3-6-3	5929	4	0	n51	195392,721	6111	4	0	n51	195392,721
3-6-4	6010	4	0	n51	199978,84	6427	5	0	n51	199978,84
3-6-5	5886	4	0	n51	195184,897	6404	5	0	n51	195184,897
Average	5880,2	4	0		196750,8152	6241,6	4,4	0		196750,8152

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-7-1	10800	9,3	n51 n52	206860,133	5158	4	0	n52 n53	194986,757
3-7-2	10800	7,08	n51 n52	200703,168	5081	4	0	n52 n53	194804,702
3-7-3	10800	6,62	n51 n52	199032,776	4937	4	0	n52 n53	193244,566
3-7-4	10800	9,18	n51 n52	205812,611	5131	4	0	n52 n53	197058,717
3-7-5	10800	5,71	n51 n52	201632,674	5105	4	0	n52 n53	198717,491
Average	10800	7,578		202808,2724	5082,4	4	0		195762,4466

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-7-1	4794	4	0	n52 n53	194986,757	5722	5	0	n52 n53	194986,757
3-7-2	4771	4	0	n52 n53	194804,702	5521	5	0	n52 n53	194804,702
3-7-3	4851	4	0	n52 n53	193244,566	5712	4	0	n52 n53	193244,566
3-7-4	4792	4	0	n52 n53	197058,717	5603	4	0	n52 n53	197058,717
3-7-5	4794	4	0	n52 n53	198717,491	5482	4	0	n52 n53	198717,491
Average	4800,4	4	0		195762,4466	5608	4,4	0		195762,4466

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-8-1	10800	10,44	n51 n52	206431,932	5785	4	0	n52 n53	194415,295
3-8-2	10800	10,9	n51 n52	207643,908	5566	4	0	n52 n53	193774,136
3-8-3	10800	11,97	n51 n52	209328,277	5769	4	0	n52 n54	193326,285
3-8-4	10800	11,23	n51 n52	207563,03	5741	4	0	n52 n54	194093,55
3-8-5	10800	10,07	n51 n52	206803,386	5645	4	0	n52 n54	194918,661
Average	10800	10,922		207554,1066	5701,2	4	0		194105,5854

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-8-1	5460	4	0	n52 n53	194415,295	6372	5	0	n52 n53	194415,295
3-8-2	5470	4	0	n52 n53	193774,136	6519	5	0	n52 n53	193774,136
3-8-3	5528	4	0	n52 n54	193326,285	6404	5	0	n52 n54	193326,285
3-8-4	5538	4	0	n52 n54	194093,55	6465	4	0	n52 n54	194093,55
3-8-5	5473	4	0	n52 n54	194918,661	6358	4	0	n52 n54	194918,661
Average	5493,8	4	0		194105,5854	6423,6	4,6	0		194105,5854

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-9-1	10800	16,26	n51 n52	220163,138	6035	4	0	n52 n53	192924,42
3-9-2	10800	15,18	n51 n52	217414,21	6039	5	0	n52 n53	193760,776
3-9-3	10800	16,29	n51 n52	218835,71	6120	4	0	n52 n54	192914,764
3-9-4	10800	15,04	n51 n52	216647,18	6183	4	0	n52 n54	193145,266
3-9-5	10800	9,56	n51 n52	204921,925	6083	4	0	n52 n54	193813,366
Average	10800	14,466		215596,4326	6092	4,2	0		193311,7184

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-9-1	5879	4	0	n52 n53	192924,42	6995	4	0	n52 n53	192924,42
3-9-2	5930	4	0	n52 n53	193760,776	6623	4	0	n52 n53	193760,776
3-9-3	5945	4	0	n52 n54	192914,764	6758	5	0	n52 n54	192914,764
3-9-4	5903	4	0	n52 n54	193145,266	6696	5	0	n52 n54	193145,266
3-9-5	5862	4	0	n52 n54	193813,366	6736	5	0	n52 n54	193813,366
Average	5903,8	4	0		193311,7184	6761,6	4,6	0		193311,7184

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-10-1	10800	6,81	n51 n53	199955,297	5111	4	0	n53 n54	194860,379
3-10-2	10800	8,19	n52 n53	202797,828	5093	4	0	n51 n54	195139,381
3-10-3	10800	8,15	n52 n53	204639,515	5288	5	0	n51 n54	196447,934
3-10-4	10800	7,65	n51 n54	204077957	5093	4	0	n51 n52	195850,842
3-10-5	10800	5,86	n51 n52	202218,566	5054	4	0	n51 n52	198865
Average	10800	7,332		40977513,64	5127,8	4,2	0		196232,7072

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-10-1	4881	4	0	n53 n54	194860,379	5835	5	0	n53 n54	194860,379
3-10-2	5048	4	0	n51 n54	195139,381	5926	4	0	n51 n54	195139,381
3-10-3	4919	4	0	n51 n54	196447,934	6056	5	0	n51 n54	196447,934
3-10-4	4852	4	0	n51 n52	195850,842	5751	5	0	n51 n52	195850,842
3-10-5	4985	4	0	n51 n52	198865	5721	5	0	n51 n52	198865
Average	4937	4	0		196232,7072	5857,8	4,8	0		196232,7072

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-11-1	10800	10,04	n51 n54	204881,054	5670	4	0	n51 n52	192990,792
3-11-2	10800	12,69	n52 n53	210902,67	5734	4	0	n51 n52	193345,989
3-11-3	10800	13,74	n52 n53	213972,28	5676	4	0	n53 n54	194369,523
3-11-4	10800	10,91	n51 n52	206761,074	5771	5	0	n53 n54	194579,462
3-11-5	10800	9,91	n51 n54	208741,894	5666	5	0	n51 n52	197228,539
Average	10800	11,458	#SAYI/0!	209051,7944	5703,4	4,4	0		194502,861

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-11-1	5467	4	0	n51 n52	192990,792	6398	5	0	n51 n52	192990,792
3-11-2	5382	4	0	n51 n52	193345,989	6144	5	0	n51 n52	193345,989
3-11-3	5547	4	0	n53 n54	194369,523	6400	5	0	n53 n54	194369,523
3-11-4	5397	4	0	n53 n54	194579,462	6423	5	0	n53 n54	194579,462
3-11-5	5397	4	0	n51 n52	197228,539	6527	4	0	n51 n52	197228,539
Average	5438	4	0		194502,861	6378,4	4,8	0		194502,861

	PSNDP				PSNDP-B1				
Set No	CPU (s)	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-12-1	10800	12,99	n53 n54	211314,949	6065	4	0	n51 n52	192415,695
3-12-2	10800	15,45	n51 n53	216812,72	6138	5	0	n51 n52	192860,014
3-12-3	10800	14,43	n51 n53	215527,93	6012	4	0	n53 n54	193790,91
3-12-4	10800	14,58	n51 n53	213542,818	6253	5	0	n53 n54	192346,863
3-12-5	10800	15,63	n51 n52	222158,462	6076	4	0	n51 n52	196472,214
Average	10800	14,616		215871,3758	6108,8	4,4	0		193577,1392

	PSNDP-B2					PSNDP-B3				
Set No	CPU (s)	# iter.	Gap (%)	storage opened	Opt.	CPU (s)	# iter.	Gap (%)	storage opened	Opt.
3-12-1	5869	4	0	n51 n52	192415,695	6859	5	0	n51 n52	192415,695
3-12-2	5950	4	0	n51 n52	192860,014	6897	5	0	n51 n52	192860,014
3-12-3	5805	4	0	n53 n54	193790,91	6918	5	0	n53 n54	193790,91
3-12-4	5833	4	0	n53 n54	192346,863	6911	5	0	n53 n54	192346,863
3-12-5	5839	4	0	n51 n52	196472,214	6984	5	0	n51 n52	196472,214
Average	5859,2	4	0		193577,1392	6913,8	5	0		193577,1392

B. APPENDIX – B STOCHASTIC MODEL INDIVIDUAL RESULTS

Table B.1. *Stochastic Model Individual Results*

Set No	CPU (s)	# of iteration	Gap(%)	Cost
1-1	10490,00	8	0	88912,46
1-2	10158,00	9	0	89348,30
1-3	10685,00	9	0	89348,30
1-4	10533,00	8	0	84206,19
1-5	10570,00	9	0	83424,98
1-6	10800,00	-	1,47	-
1-7	10800,00	-	1,91	-
1-8	10800,00	-	2,32	-
1-9	10800,00	-	2,44	-
1-10	10800,00	-	2,84	-
1-11	10800,00	-	3,35	-
1-12	10800,00	-	4,77	-

Set No	CPU (s)	# of iteration	Gap(%)	Cost
2-1	10398,00	8	0	80246,43
2-2	10481,00	8	0	79569,69
2-3	10675,00	9	0	79269,68
2-4	10752,00	9	0	81497,49
2-5	10536,00	9	0	81386,52
2-6	10800,00	-	1.51	-
2-7	10145,00	9	0	83302,55
2-8	10250,00	9	0	81933,95
2-9	10800,00	-	5,96	-
2-10	10800,00	-	2,85	-
2-11	10800,00	-	4,70	-
2-12	10800,00	-	5,91	-

Set No	CPU (s)	# of iteration	Gap(%)	Cost
3-1	10564,00	7	0	79552,64
3-2	10376,00	8	0	79461,73
3-3	10590,00	10	0	79289,12
3-4	10783,00	9	0	80924,38
3-5	10397,00	9	0	80924,38
3-6	10800,00	-	2,41	-
3-7	10788,00	8	0	82130,30
3-8	10800,00	-	2,46	-

3-9	10800,00	-	4,05	-
3-10	10800,00	-	4,47	-
3-11	10800,00	-	4,58	-
3-12	10800,00	-	4,78	-

Set No	CPU (s)	# of iteration	Gap(%)	Cost
4-1	10064,00	8	0	80128,44
4-2	10428,00	9	0	79516,91
4-3	10768,00	10	0	79489,16
4-4	10179,00	9	0	79013,76
4-5	10667,00	9	0	79013,76
4-6	10800,00	-	2,17	-
4-7	10800,00	-	2,77	-
4-8	10800,00	-	3,17	-
4-9	10800,00	-	3,49	-
4-10	10800,00	-	4,44	-
4-11	10800,00	-	4,51	-
4-12	10800,00	-	4,53	-

Set No	CPU (s)	# of iteration	Gap(%)	Cost
5-1	10249,00	9	0	81299,62
5-2	10520,00	9	0	81272,50
5-3	10646,00	9	0	81272,50
5-4	10453,00	9	0	79702,75
5-5	10593,00	9	0	79512,75
5-6	10800,00	-	1,81	-
5-7	10113,00	9	0	82330,55
5-8	10517,00	10	0	81533,48
5-9	10800,00	-	2,68	-
5-10	10800,00	-	2,93	-
5-11	10800,00	-	3,57	-
5-12	10800,00	-	5,57	-

C. APPENDIX – C STOCHASTIC TURKISH PIPELINE NETWORKS

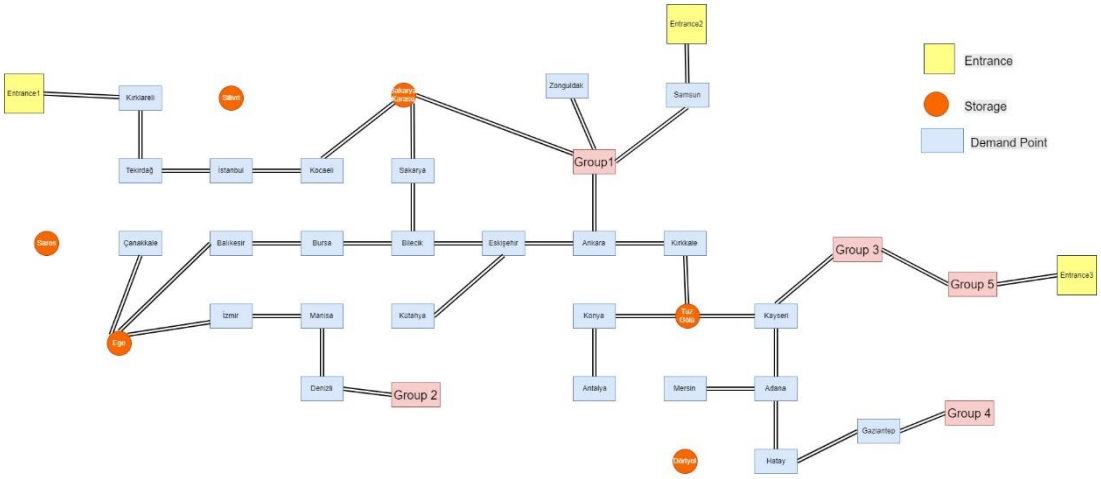

Figure C.1. Case 1 Network Structure

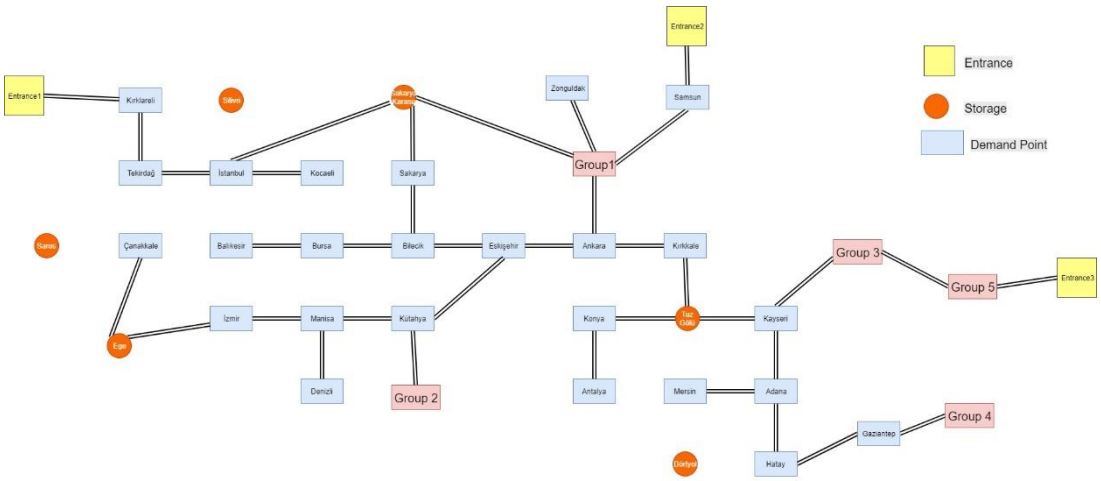

Figure C.2. Case 2 Network Structure

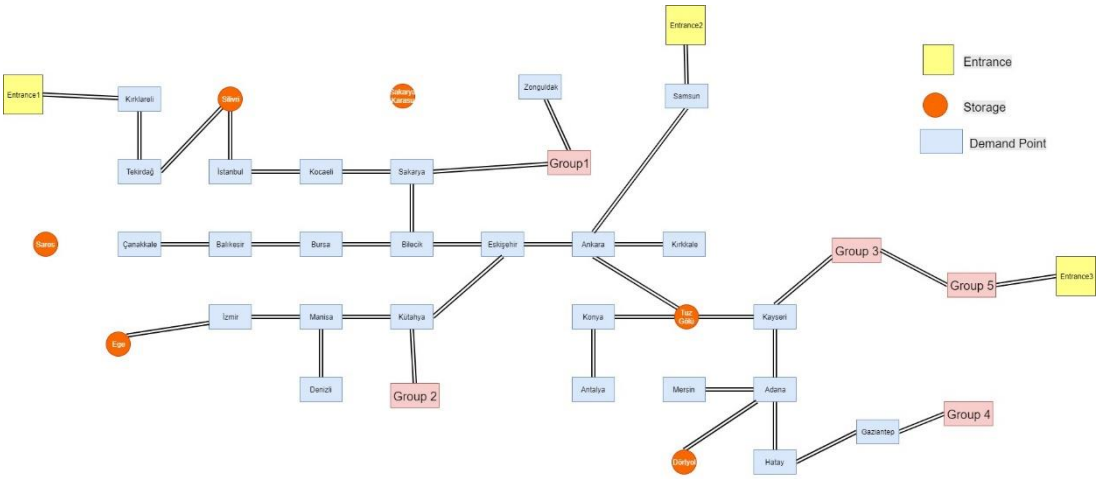

Figure C.3. Case 3 Network Structure

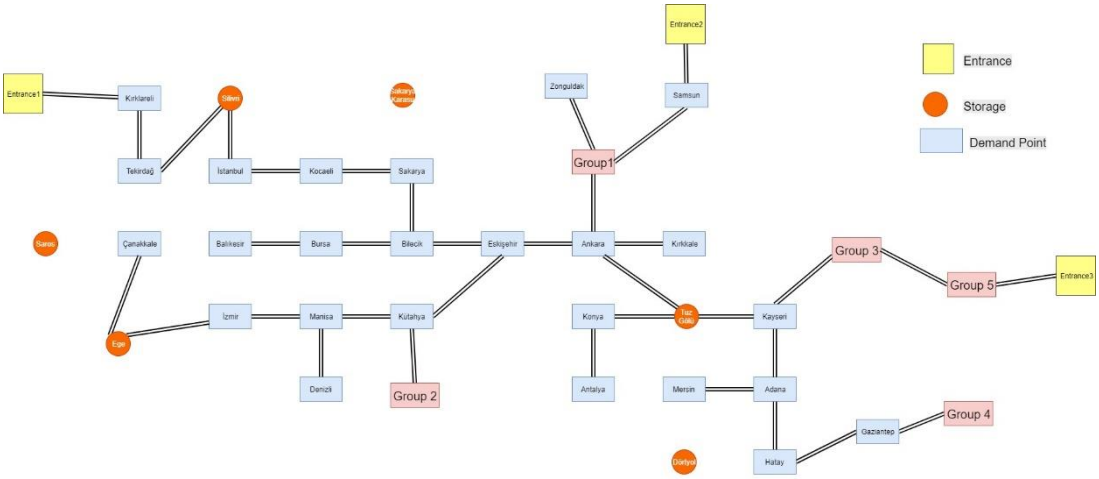

Figure C.4. Case 4 Network Structure

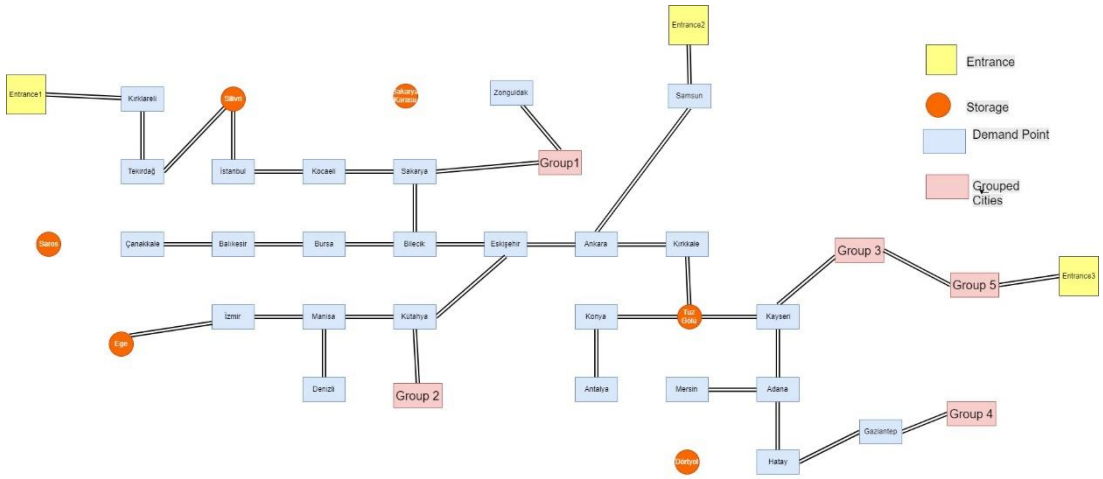

Figure C.5. Case 5 Network Structure

D. APPENDIX PIPELINE FLOW EQUATIONS

The pipeline flow and diameter relationship is analyzed to get rid of second order equations in the model. Edgar et al. (1978) shows that pipe flow capacity is calculated with Equation D.1.

$$capacity(arc) = \frac{871 * 10^{-6} * d_{arc}^{\frac{8}{3}} * (Pd(i)^2 - Ps(i)^2)}{\sqrt{l(arc)}} \quad \text{(D.1)}$$

where;

d_{arc} : Diameter of pipeline (inches)

$Pd(i)$: Discharge pressure of pipeline segment (psi)

$Ps(i)$: Suction pressure of pipeline segment (psi)

l_{arc} : Length of pipeline segment

Flow and diameter relationship is generated for set of length, discharge and suction pressure values. Trend line equation slope value in Figure D.1 is used in the mathematical model to present the relationship.

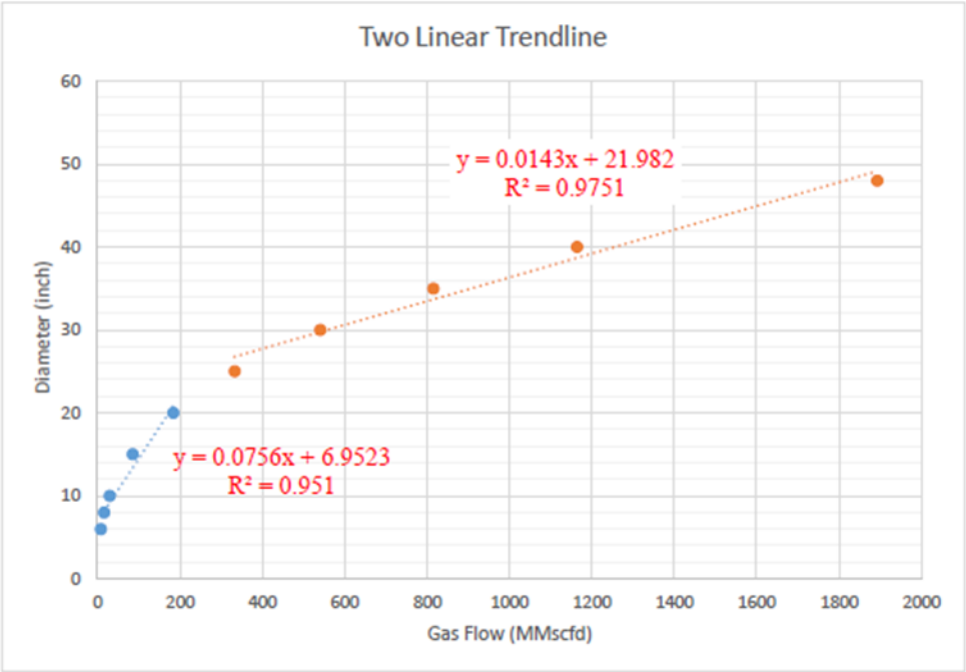

Figure D.1. Relationship between Pipe Diameter and Gas Flow Rate