

An Improved Mathematical Formulation
For the Carbon Capture and Storage (CCS) Problem

by

Loy Joseph Lobo

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2017 by the
Graduate Supervisory Committee:

Jorge Sefair, Co-Chair
Adolfo Escobedo, Co-Chair
Michael Kuby

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

Carbon Capture and Storage (CCS) is a climate stabilization strategy that prevents CO₂ emissions from entering the atmosphere. Despite its benefits, impactful CCS projects require large investments in infrastructure, which could deter governments from implementing this strategy. In this sense, the development of innovative tools to support large-scale cost-efficient CCS deployment decisions is critical for climate change mitigation. This thesis proposes an improved mathematical formulation for the scalable infrastructure model for CCS (*SimCCS*), whose main objective is to design a minimum-cost pipe network to capture, transport, and store a target amount of CO₂. Model decisions include source, reservoir, and pipe selection, as well as CO₂ amounts to capture, store, and transport. By studying the *SimCCS* optimal solution and the subjacent network topology, new valid inequalities (VI) are proposed to strengthen the existing mathematical formulation. These constraints seek to improve the quality of the linear relaxation solutions in the branch and bound algorithm used to solve *SimCCS*. Each VI is explained with its intuitive description, mathematical structure and examples of resulting improvements. Further, all VIs are validated by assessing the impact of their elimination from the new formulation. The validated new formulation solves the 72-nodes Alberta problem up to 7 times faster than the original model. The upgraded model reduces the computation time required to solve *SimCCS* in 72% of randomly generated test instances, solving *SimCCS* up to 200 times faster. These formulations can be tested and then applied to enhance variants of the *SimCCS* and general fixed-charge network flow problems. Finally, an experience from testing a Benders decomposition approach for *SimCCS* is discussed and future scope of probable efficient solution-methods is outlined.

DEDICATION

To,

*My parents, Irine Lobo and Joseph Lobo, my sisters, Leera Lobo and Lora Lobo,
Lord Jesus Christ, my late cousin Melwin Lobo, friends and other family members for
their constant prayers, support, guidance and wisdom.*

ACKNOWLEDGMENTS

I would like to thank Dr. Jorge A. Sefair for giving me the opportunity to learn, research and apply operations research techniques to this very interesting problem of CCS. I have gained great belief and confidence in my abilities due to the encouragement and belief he imposed in me. Dr. Michael Kuby has inspired me by the course 'Location analysis, modelling and GIS' which beautifies the approach of integration of OR with other sciences. I am grateful to him for the massive interest I have gained in research in this field. Dr. Richard Middleton has been very supportive and I thank him and Dr. Kuby for the opportunity to work on their paper. I would also like to thank Dr. Adolfo Escobedo for being a Co-Chair. I have massively learned from his valuable feedback on my research. Finally, I would like to thank all my other colleagues for their support and wishes.

TABLE OF CONTENTS

	Page
LIST OF TABLES.....	vi
LIST OF FIGURES	vii
CHAPTER	
1.0 INTRODUCTION	1
1.1 Background	1
1.2 Literature Review.....	1
1.3 Scope	4
2.0 MATHEMATICAL MODELS.....	6
2.1 SimCCS Model	6
2.2 Strengthening the SimCCS Model with Valid Inequalities.....	9
2.2.1 Target Storage Inequality	11
2.2.2 Source Requisite Inequalities	13
2.2.3 Reservoir Requisite Inequalities	15
2.2.4 Incoming Pipe Requisite Inequalities.....	16
2.2.5 Outgoing Pipe Requisite Inequalities.....	18
2.2.6 Distinction Between Transshipment and Other Nodes	19
2.2.7 Flow Through a Transshipment Inequalities.....	19
2.2.8 Pipes Through a Transshipment Inequalities	20

CHAPTER	Page
2.2.9 Inequality For Minimum Number Of Pipes (Forest Inequalities).....	21
2.2.10 Combined Effect Of Valid Inequalities.....	23
3.0 COMPUTATIONAL EXPERIMENTS & RESULTS	26
3.1 Analysis Of Effectiveness Of Valid Inequalities.	26
3.2 Testing MIP' On a Real Problem	28
3.3 Testing MIP' On Generated Datasets.....	30
3.3.1 Computational Results	32
3.3.2 Consistent Improvement In Linear Relaxation	38
4.0 CONCLUSIONS AND FUTURE WORK.....	41
REFERENCES	43
APPENDIX	
A MIP' CONSISTING OF CONSTRAINTS ((1) TO (14), (V1) TO (V11)).....	47

LIST OF TABLES

Table	Page
1. Network Analysis	21
2. Test for Validity of VIs on MIP'	27
3. (a) Solution Times of Randomly Generated Instances – 20 and 39 Nodes	33
3. (b) Solution Times of Randomly Generated Instances -78 And 117 Nodes	34
4. Objective Function Ratio of Instances That Time-Out.....	37
5. (a) Linear Relaxation Index of Randomly Generated Instances – 20 & 39 Nodes	39
5. (b) Linear Relaxation Index of Randomly Generated Instances – 78 & 117 Nodes	40

LIST OF FIGURES

Figure	Page
1. (a) Complete CCS Network with 3 Pipe Choices Per Arc (b) Optimal CCS Solution...	8
2. Illustration of A Valid Inequality For The Feasible Region of Constraints(C1), (C2), (C3), (VI1), (VI2), (VI3).....	10
3. Example Representing Benefits of Adding (V1)	12
4. Example Representing Benefits of Adding (V2)	14
5. Example Representing Benefits of Adding (V3)	15
6. Example Representing Benefits of Adding (V4)	17
7. Example Representing Benefits of Adding (V5)	18
8. Example Representing Benefits of Adding (V6), (V7)	22
9. Example Representing Benefits of Adding (V10) to Combination of Constraints	24
10. Example Representing Benefits of Adding (V11) to Combination of Constraints	25
11. Map of Oil Sands in Alberta	29
12. Alberta Network for <i>SimCCS</i>	29
13. Sparse Network of 5 Nodes, 5 Reservoirs and 10 Transshipments	31
14. Dense Network of 5 Nodes, 5 Reservoirs and 10 Transshipments.....	31
15. Scatter Diagram of Speed-Up vs Number of Pipes	36
16. Scatter Diagram of Speed-Up vs Percentage of Max-Flow.....	36
17. Proximity of Linear Relaxation to An Optimal Solution.....	38

1.0 INTRODUCTION

1.1 Background

Climate change through accumulating greenhouse gases is a threat to life on Earth. Carbon dioxide contributes hugely to global warming and if emissions are not controlled, natural processes of removing it will not be sufficient to stop its accumulation [1]. The increasing use of conventional sources of energy in developing countries means that every emission reduction measure would be important [2]. Substituting towards nickel farming, use of biodiesel for fuel, use of alternative energy vehicles, and prevention of forest fires, are some of the ways to reduce emissions [3], but acceleration of the use of carbon capture and storage (CCS) is important to approach the global target of limiting temperature-rise to 1.5 ° C [4]. Impactful CCS projects require large investments in infrastructure (the US alone accounts for more than 7,600 km of pipelines [4]), which could deter governments from using CCS. In this sense, the development of innovative tools to support large-scale cost-efficient CCS deployment decisions is critical to continue contributing to climate mitigation.

1.2 Literature review

CCS refers to the integrated infrastructure to capture CO₂ from high-emission sources, its safe transport to a dedicated storage reservoir and subsequent sequestration from the atmosphere [5]. Its global status is 38 large-scale approved projects around the world [4], and reduction in unit investment costs in this technology would be realized with

economies of scale. Minimizing the overall cost of CCS is a challenging problem involving decisions of determination of sources, sinks, and pipes, and the amount of storage, capture and flow of CO₂ through this network [6]. Being a mixed integer programming problem, solving the CCS for large-scale instances with commercial solvers becomes difficult and computationally time-consuming [7]. This suggests that improving the existing mathematical formulations and providing new solution algorithms will be determinant in solving real CCS instances.

CCS includes decisions for source-sink-pipeline-network design and network flows. Amongst these decisions, pipeline network design and flow problems have been previously studied for other gases. One of the early papers on natural gas pipeline systems referred to a model consisting of optimizing a network of sources, sinks and pipeline diameters [8]. Decisions included selection of pipe diameters, but they did not consider the selection of sources and reservoirs. Another paper on optimal pipeline network design for a gas described networks as connected trees and involved selection of pipe diameters and junction nodes [9]. In this case also, decisions did not include a complete network design. More recently, a research on supply chain network design for packaged gases included a mathematical model consisting of optimal selection of: hubs as transshipment nodes, filling plants as sources, flow on every arc [10]. This was a multi-product level mixed integer program and a decomposition approach was employed to solve it efficiently.

For CCS infrastructure optimization, an initial study hinted at planning a combination of ships and pipelines for CO₂ transport [11]. Related to pipeline network design for CCS,

Dooley et al. (2006) developed a cost-minimization formulation consisting of basic network elements in a CCS network [12], but it did not consider realistic network features in terms of spatial arrangement. Another CCS infrastructure model involved the development of a source-sink matching algorithm named “String of Pearls”, but it considered direct straight-line pipeline networks between sources and sinks [13]. In this respect, the scalable infrastructure model (*SimCCS*) [6] was a more sophisticated formulation in terms of network configuration, spatial arrangement, decision of pipeline-diameter-on-arc, and complete system-optimal applications. Variants of this model in the form of *SimCCS^{PRICE}* [7] and *SimCCS^{TIME}* [14] were developed to incorporate objectives considering carbon-tax and temporal requirements respectively. Efforts to improve *SimCCS* scalability led to a model involving linearization of pipeline diameters [15], in which the decision of pipes with discrete capacities is substituted with that of continuous capacities. *SimCCS* has also inspired models for optimization of hydrogen supply chain infrastructure [16] and wind-power generation infrastructure in SimWIND [17].

From the available CCS models, *SimCCS* is one of the most realistic applications in terms of integrated networks, commercially available pipeline diameters and system-optimal solutions. An enhancement in this model would be a first step towards improving all CCS models and increase their use by decision-makers. Hence, we selected the basic *SimCCS* model for researching improvements in solution times.

SimCCS is a variant of the fixed-charge network flow (FCNF) problem: it involves the selection of sources, reservoirs, transshipment nodes, pipes, pipeline network design, and network flow, for carbon capture and storage [6]. Solving an FCNF, which is a mixed

integer programming (MIP) problem, takes long times for large scale instances when commercial solvers are used [18]. With this thesis, we aim to solve the *SimCCS* problem more efficiently. Regarding the solution methods for FCNF and pipeline-network design problems, a myriad of exact [23, 25, 29, 30, 31,32, 36], heuristic [24, 26-28, 33-34, 37, 38], and hybrid [35] approaches are available in the literature. To achieve efficient solutions, we adopt the exact approach (to obtain optimal solutions) of using valid inequalities to exploit the variant structure of the CCS problem via a tighter formulation.

An early research on general fixed charge problems developed valid inequalities for generating cutting planes to improve the formulations [39]. Another study on capacitated fixed charge flow problems discusses valid inequalities and examines linear relaxation to generate an efficient-solution heuristic [40]. While there has been existing work on valid inequalities on capacitated fixed-charge network flow [19] and fixed-charge multicommodity network design problems [20], their structure does not involve the selection-decision of a single pipe on every arc from a set of available diameters. Also, the flexibility for the *SimCCS* model to decide different diameters of pipes on consecutive arcs on a path makes the problem more complex. In this sense, our proposed approach of valid inequalities would lead to discovery of novel constraints.

1.3 Scope

We study the scalable infrastructure model for carbon capture and storage (*SimCCS*), whose mathematical model seeks to minimize the overall cost of sequestration, under sequestration, transport and storage of CO₂ constraints [6]. By studying the *SimCCS*

solution and the properties of the subjacent network topology, we propose new valid inequalities to strengthening the existing mathematical formulation. These constraints improve the root node solution (linear relaxation) of the branch and bound algorithm used for *SimCCS*. For each valid inequality (VI), we provide an intuitive description, mathematical structure, and examples illustrating their improvements on the linear relaxation. Further, we validate the VIs by assessing the impact of removing them from the new formulation (CCS coupled with VIs). We discuss the results of testing our formulation on a real large-scale problem and on randomly generated instances.

Our approach yields improvement in computational times of CCS solutions for 72% of the tested instances, reaching speedups of up to 200 times compared to *SimCCS*. We also discuss the instance structure where our inequalities are most impactful and outline their future application to variants of the *SimCCS* and general fixed-charge network flow problems. Finally, we describe our experience from testing a Benders decomposition approach for *SimCCS*.

2.0 MATHEMATICAL MODELS

The goal of this chapter is to describe mathematical models of existing CCS problem and of our proposed formulation. We explain the structure of the formulation with its intuition, constraints and illustrations.

2.1 *SimCCS* model

The CCS problem is to design a network able to transport a target amount of CO₂—denoted by τ (T in Middleton et al. [6])—from sources to reservoirs. The network consists of source nodes (S), reservoir nodes (R), arcs representing pipes (A), and transshipment nodes denoting pipe intersections (T). (N_i^+) denotes the set of nodes adjacent to node ‘i’ but accessible using arcs starting from node ‘i’. (N_i^-) denotes the set of nodes adjacent to node ‘i’ but accessible using arcs ending at node ‘i’. Sources and reservoirs are subject to maximum capture and storage capacities, denoted by Q_i^S and Q_j^R for source i and reservoir j , respectively. Similarly, F_i^S and V_i^S denote the fixed cost (land purchase, construction, and technology installation) and variable operational cost (pumping and maintenance) for source i , whereas parameters F_j^R and V_j^R represent the fixed and variable costs for reservoir j , respectively. The minimum and maximum capacity of arc (i, j) —denoted by Q_{ijd}^{pmin} and Q_{ijd}^{pmax} (mentioned as $\min Q_{ijd}^{\text{p}}$ and $\max Q_{ijd}^{\text{p}}$ in Middleton et al. [6])—depend on the chosen pipe diameter $d \in D$, where D is the set of commercially available diameters. Using a pipe incurs in a variable operational cost (V_{ij}^{p}) per ton of CO₂ transported, and a fixed-charge construction cost (F_{ijd}^{p}). Decision variables include the amount of CO₂ that is captured at source i (a_i) and stored at

reservoir j (b_j), as well as the amount transported on arc (i, j) (x_{ij}). Additional variables represent network design decisions, including operating or not source i (s_i) and reservoir j (r_j), and constructing or not a pipe of diameter d on arc (i, j) (y_{ijd}).

$$\text{Minimize } \sum_{i \in S} (F_i^s s_i + V_i^s a_i) + \sum_{i \in I} \sum_{j \in N_i^+} \sum_d F_{ijd}^p y_{ijd}$$

$$+ \sum_{i \in I} \sum_{j \in N_i^+} V_{ij}^p x_{ij} + \sum_{j \in R} (F_j^r r_j + V_j^r b_j) \quad (1)$$

$$x_{ij} - \sum_{d \in D} Q_{ijd}^{\text{pmax}} y_{ijd} \leq 0, \forall i \in I, j \in N_i^+ \quad (2)$$

$$x_{ij} - \sum_{d \in D} Q_{ijd}^{\text{pmin}} y_{ijd} \geq 0, \forall i \in I, j \in N_i^+ \quad (3)$$

$$\sum_{j \in N_i^+} x_{ij} - \sum_{j \in N_i^-} x_{ji} - a_i + b_i = 0, \forall i \in I \quad (4)$$

$$a_i - Q_i^s s_i \leq 0, \forall i \in S \quad (5)$$

$$b_j - Q_j^r r_j \leq 0, \forall j \in R \quad (6)$$

$$\sum_{i \in S} a_i \geq \tau \quad (7)$$

$$\sum_{d \in D} y_{ijd} \leq 1, \forall i \in I, j \in N_i^+ \quad (8)$$

$$y_{ijd} \in \{0, 1\}, \forall i \in I, j \in N_i^+, d \in D \quad (9)$$

$$s_i \in \{0, 1\}, \forall i \in S \quad (10)$$

$$r_j \in \{0, 1\}, \forall j \in R \quad (11)$$

$$x_{ij} \geq 0, \forall i \in I, j \in N_i^+ \quad (12)$$

$$a_i \geq 0, \forall i \in S \quad (13)$$

$$b_j \geq 0, \forall j \in R \quad (14)$$

The CCS problem is written in equations (1) to (14). The constraints consist on flow-balance conditions, minimum and maximum arc flow capacity, CO₂ target capture, and logical conditions specifying that capture and storage is only possible in sources and reservoirs that are operating, and that at most one pipe is selected for each arc. The objective function is to minimize the total cost, consisting of variable and fixed-charge components. Figure 1(a) shows an example of a CCS instance with three candidate pipes for each arc, and Figure 1(b) shows its optimal solution.

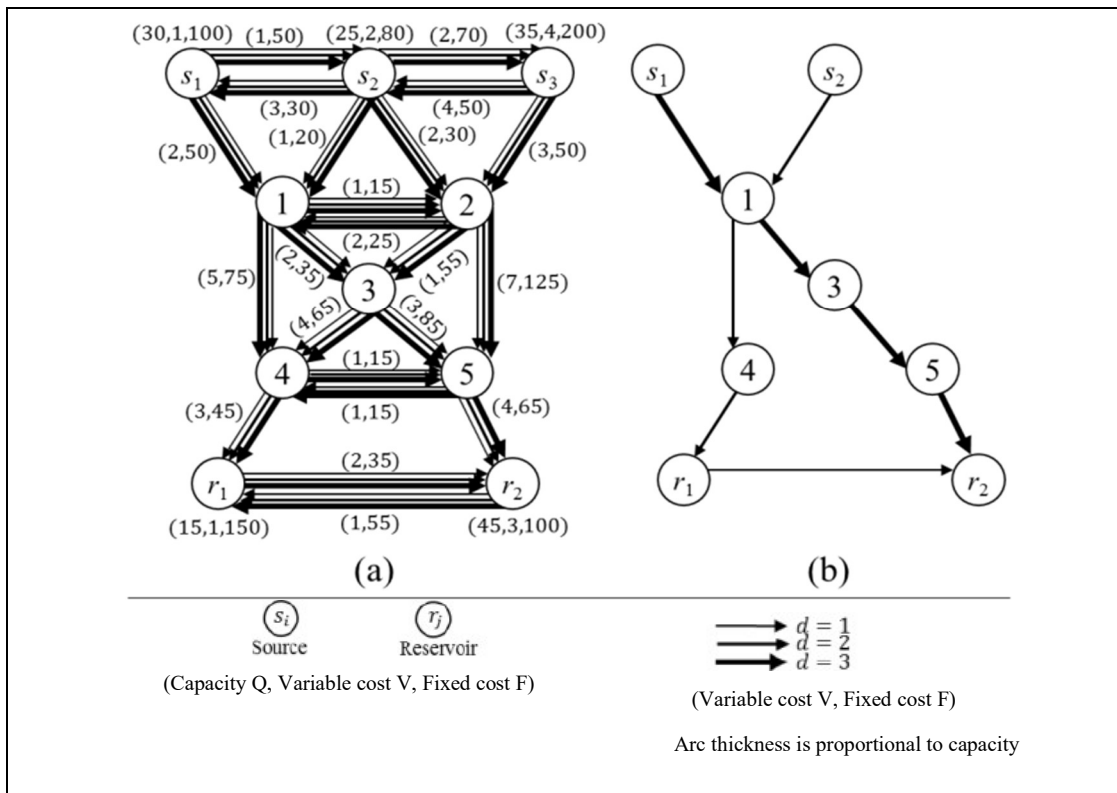


Figure 1. (a) Complete CCS network with three pipe choices per arc. (b) Optimal CCS solution.

2.2 Strengthening the *SimCCS* model with Valid Inequalities

The *SimCCS* model is a capacitated fixed charge network flow (FCNF) problem that contains continuous flow variables representing the amount of CO₂ transported and binary decision variables for the selection of pipe capacities. Depending on the solver, this mixed integer programming (MIP) model is generally solved using a branch-and-bound algorithm aided with presolve, cutting planes, parallelism, and heuristics [41]. The root node of the branch-and-bound tree is a solution with relaxation of integrality constraints typically obtained as the MIP's linear programming (LP) relaxation. Strengthening of this initial solution results in a tighter feasible space and fewer subsequent branches towards the optimal solution. This thesis focuses on improving the optimal value of the LP relaxation via identification of tighter valid inequalities as constraints. To produce such inequalities, we examined fractional solutions of the root node where the structure of the resulting network revealed potential directions of improvement. This analysis of the structure of the optimal network configuration over multiple datasets for CCS problem revealed the relationship between linear flow variables and binary network-design variables. We derived several rules of a network flow to tighten the formulation, producing valuable constraints. Cuts based on inherent characteristics of pipe network design yielded an improved solution set of fractional values of the integer-variables - s , r and y . These constraints tighten the feasible region while ensuring any integer solution from the feasible region is not cut.

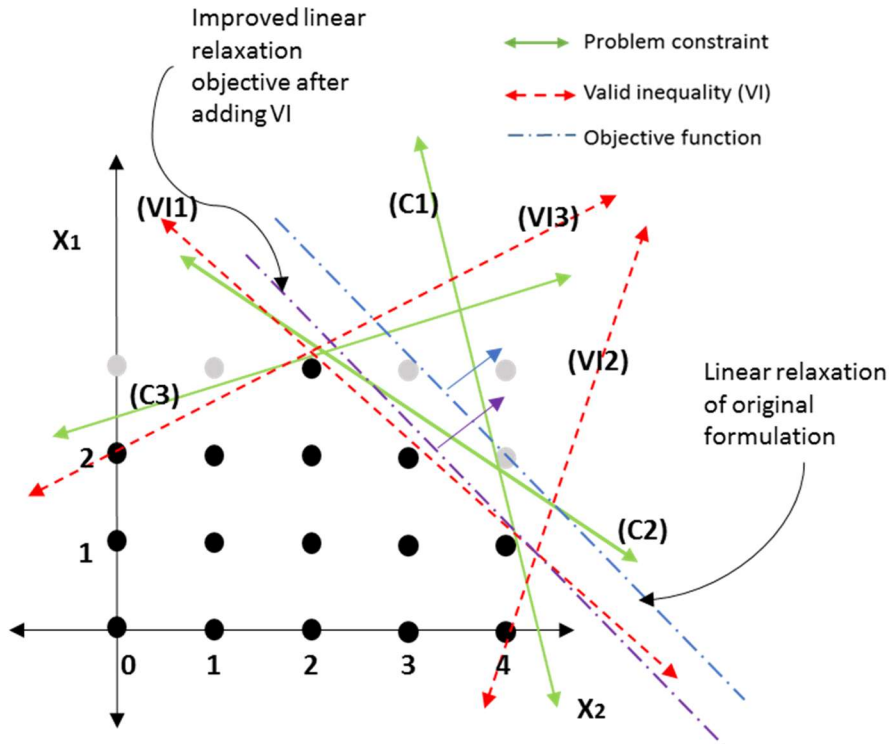


Figure 2: Illustration of a valid inequality for the feasible region of constraints(C1), (C2), (C3), (VI1), (VI2), (VI3) in a two-variable multi-constraint integer problem

Figure 2 illustrates the feasible region for a two-variable multi-constraint integer problem. The feasible space for this integer programming problem is analogous to the MIP of *SimCCS* with binary variables. The dark dots denote feasible integer solutions, which are not eliminated by any of the valid inequality constraint lines.

The addition of sets of valid constraints may result in improved root relaxations thus reducing the number of iterations towards the solution of the MIP branch-and-bound tree [21]. In this section, we describe families of valid inequalities including their intuition, modelling constraints and examples. The individual and combined effects of these inequalities are studied in section 3.0. The *SimCCS* problem with all valid inequality constraints added is called *MIP'*.

2.2.1 Target storage inequality

This inequality is inspired by two observations. First, the amount of CO₂ captured in the CCS system cannot remain in the network without being stored. Second, the capacity of the open reservoirs should be enough to store the target CO₂. Constraint (V1) forces target amount of CO₂ sequestration to be less than the sum of products of reservoir capacities and reservoir-opening binary variables. This drives the binaries to have higher fractional values compared to *SimCCS* resulting in a tighter linear relaxation.

$$\tau \leq \sum_{j \in R} Q_j^r r_j \quad (V1)$$

Network diagrams from Figure 3 to Figure 11 in this section consist of nodes represented by circles labelled with a letter indicating the node type and the number. Nodes labelled with ‘T’ are meant exclusively for transshipment. Nodes marked with ‘S’ could either act as a source or transshipment while those with ‘R’ are reservoirs or transshipments. Solid arrows represent selected arcs and they are labelled with the corresponding binary variable. For instance, $y_{T5, T6, 8}$ is the binary variable representing the selection of 8” pipe between nodes T5 and T6. Multiple labels on the same arc represent different pipes selected on that arc. Dashed arrows and light dotted circular blocks denote inactive arcs and inactive nodes, respectively. No labels on these circles indicate that the fractional value of these binaries in the LP relaxed solution is 0.

Figure 3 illustrates an improved LP relaxation from \$361.29 to \$471.06 owing to the fractional value of binary r_j for reservoir R1 being forced up from 0.19 to 1.

This was tested on a network of 3 sources, 1 reservoir and 5 transshipment nodes with 3 diameters to select on the existing arcs. The nodes and arcs are labelled with the

fractional values of binaries if they are activated. While evaluating this constraint for the relaxed LP solution of *SimCCS*, LHS = $\tau = 1.017$ MT (megatons) and RHS = $\sum_{j \in R} Q_j^r r_j = 1.012$ MT. Hence, the *SimCCS* LP relaxation solution of \$361.29 is infeasible in (V1) and the value of RHS is driven up. For *SimCCS* with (V1), RHS = $\sum_{j \in R} Q_j^r r_j = 5.086$ megatons. No integer solution is cut by (V1) and the optimal solution for both 1(a) and 1(b) is \$497.95.

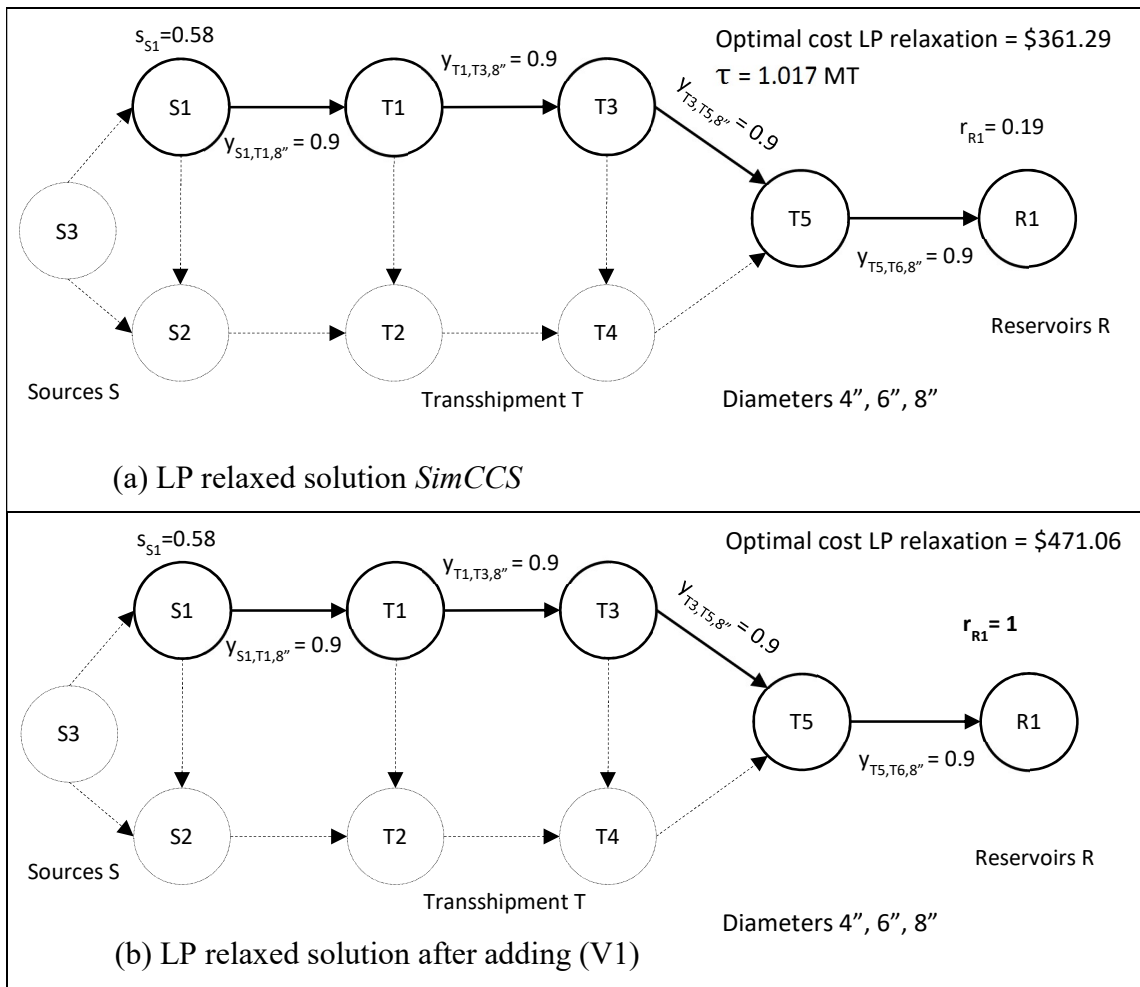


Figure 3. Example representing benefits of adding (V1)

2.2.2 Source requisite inequalities

Building any source in the *SimCCS* problem involves capture of CO₂ and this activates the corresponding s_i binary variable to 1. The captured CO₂ cannot remain in s_i and must be transported by building a pipe out or must be stored at the same node by opening a reservoir. This information leads to development of constraint (V2) by connecting source and pipe selection binary variables. Intuitively, (V2) states that: For a node to be opened as a source, at least one outgoing pipe is either built or the node must also be opened as a reservoir.

$$s_i \leq \sum_d \sum_{j \in N_i^+} y_{ijd} + r_i, \forall i \in I \quad (V2)$$

Figure 4 explains the improvement of the optimal LP relaxation from \$655.59 to \$733.94 by adding (V2). The *SimCCS* LP relaxation is infeasible for (V2) because of the binaries associated with the sources. In this case, the binary variable corresponding to source ‘S3’ is equal to 1 (LHS = $s_i = 1$) which is greater than RHS of 0.01 ($\sum_d \sum_{j \in N_i^+} y_{ijd} + r_i = 0.01 + 0$). Constraint (V2) forces the construction of pipes $y_{(S3, S1, 42'')}$ and $y_{(S3, S2, 4'')}$ with values of 0.98 and 0.02. The improved LP relaxation has a cost of \$733.939 which is closer to the optimal solution of \$1131.04 than the LP relaxation from *SimCCS*.

2.2.3 Reservoir requisite inequalities

Similar to the source requisite inequalities, building a reservoir in the *SimCCS* problem involves storage of CO₂ and this activates the corresponding r_i binary variable to 1. This captured CO₂ cannot occur without being transported by building a pipe into it or being captured at the same node by opening it as a source. Constraints (V3) specify the relationship between the corresponding reservoir and pipe selection. Intuitively, for a node to be opened as a reservoir, at least one incoming pipe is required or the node must also be opened as a source.

$$r_i \leq \sum_d \sum_{j \in N_i^-} y_{jid} + s_i, \forall i \in I \quad (V3)$$

Figure 5 illustrates the improvement of the optimal LP relaxation for the CCS instance from Figure 4.a from \$655.59 to \$669.74 by adding (V3). The *SimCCS* relaxed LP solution is infeasible for (V3) because of the r_j variable of reservoir ‘R1’. In this case, $r_i = 0.62$ is greater than RHS of 0.05 ($\sum_d \sum_{j \in N_i^-} y_{jid} + s_i = 0.05 + 0$). Constraint (V3) forces the use of an additional pipe of selection variable $y_{(T5, R1, 4'')}$, with value of 0.57.

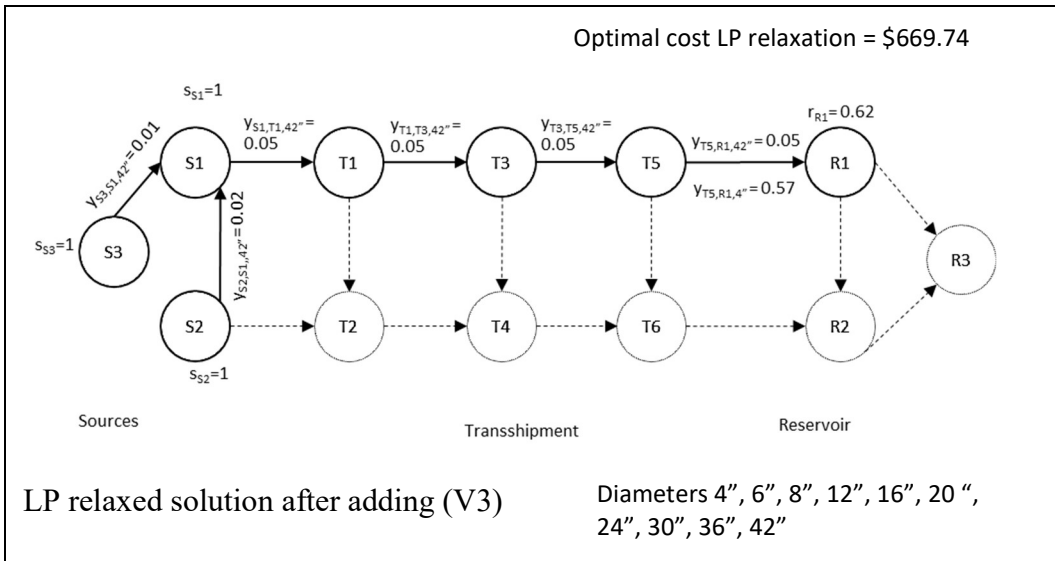


Figure 5. Example representing benefits of adding (V3)

2.2.4 Incoming pipe requisite inequalities.

The selection of any incoming pipe to a node in the *SimCCS* problem involves transportation of CO₂ and this activates the corresponding y_{ijd} binary variable to 1. This CO₂ must be transported by building a pipe out or must be stored at the same node thereby opening it as a reservoir. Constraints (V4) specify the relationship between the associated binary variables. If there is an incoming pipe to a node, at least one outgoing pipe must be built or the node must be opened as a reservoir. This valid inequality is implicitly satisfied for pipes with minimum capacities greater than 0 because of combination of constraints (3), (4), (5) and (6) in *SimCCS* (Middleton et al.,2009).

However, if such capacity is equal to zero, then the coefficient Q_{ikd}^{pmin} in constraint (3) makes the constraint redundant. Hence, (V4) is added only for the pipes with $Q_{ikd}^{pmin} = 0$.

$$y_{ikd} \leq \sum_{f \in D} \sum_{j \in N_k^+} y_{kjf} + r_k \quad , \quad i, k \in I: (i, k) \in A, d \in D : Q_{ikd}^{pmin} = 0 \quad (V4)$$

Figure 6 illustrates the improvement in the optimal LP relaxation solution \$379.43 to \$385.35 by adding (V4). The *SimCCS* relaxation LP relaxation solution in Figure 4. (a) is infeasible for (V4) because of the binaries associated with the incoming pipes to node ‘S3’. In this case, $y_{(s1, s3, 4^*)} = 0.82$ which is greater than RHS of 0 ($\sum_{f \in D} \sum_{j \in N_k^+} y_{kjf} + r_k = 0 + 0$) since there are no outgoing pipes and ‘S3’ is not selected as a reservoir.

Constraints (V4) brings force construction of many pipes by ensuring a higher fractional sum for outgoing pipes. It also changes the configuration by building additional pipes with fractional values on the same arc, opening ‘S3’ as a source, and building looping pipes between ‘S1’ and ‘S3’. This situation will be addressed later by adding additional valid inequalities.

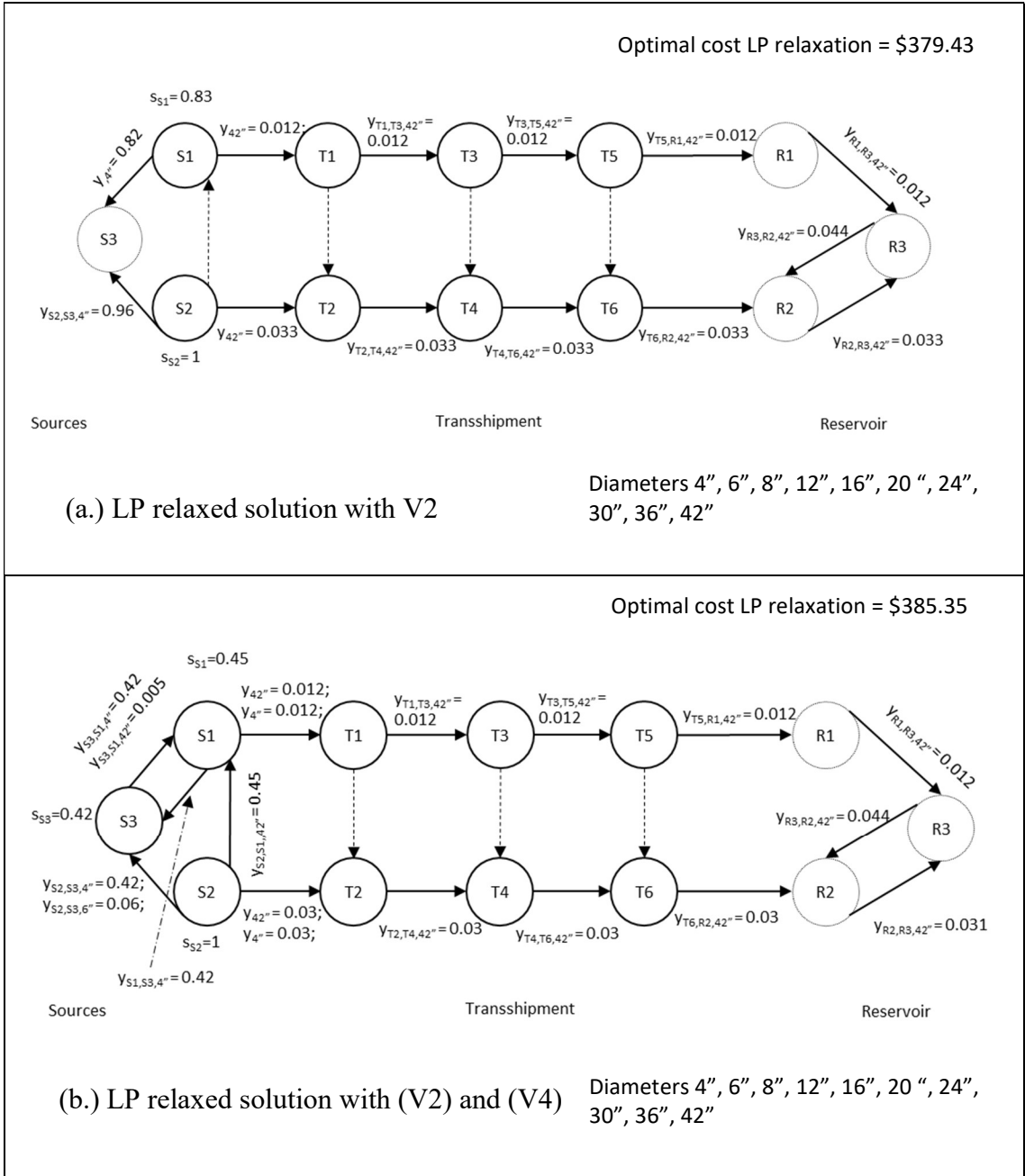


Figure 6. Example representing benefits of adding (V4)

2.2.5 Outgoing pipe requisite inequalities

Similar to the incoming pipe inequalities, selection of any outgoing pipe from a node in involves transportation of CO₂ activating the corresponding y_{ijd} binary variables to 1. This flow of CO₂ can only occur by building incoming pipes or by capturing it at the same node. Constraints (V5) specify the relationship between the associated binary variables. If an outgoing pipe from a node is to be selected, then there is at least one incoming pipe or the node must be opened as a source. This is added only for pipes with $Q_{ikd}^{pmin} = 0$ because of similar reasons as described for (V4).

$$y_{kid} \leq \sum_{f \in D} \sum_{j \in N_k^-} y_{jkf} + s_k, \forall k, i \in I: (k, i) \in A, d \in D: Q_{ikd}^{pmin} = 0 \quad (V5)$$

Figure 7 illustrates the improvement in the optimal LP relaxation solution from \$669.74 to \$674.15 by adding (V5). The *SimCCS* LP relaxation solution in Figure 5 is infeasible for (V5) because of the binaries associated with the outgoing pipe from node ‘T5’. In this case, $y_{(T5, R1, 4'')} = 0.57$ which is greater than RHS of 0.05 ($\sum_{f \in D} \sum_{j \in N_k^-} y_{jkf} + s_k = 0 + 0$). In this case, (V5) forces the activation of additional pipe selection variables.

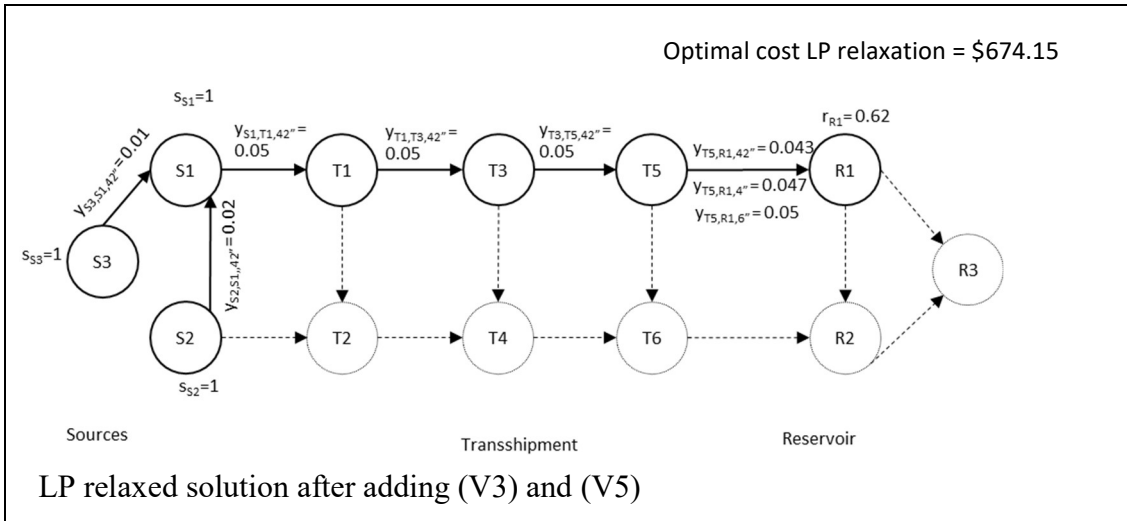


Figure 7. Example representing benefits of adding (V5)

2.2.6 Distinction between transshipment and other nodes

The subjacent network structure in *SimCCS* can be further exploited by adding variables that keep track of the transshipment node use. For this purpose, we define the binary variable t_i for every node i to specify if a node is activated as a transshipment. Because sources and reservoir nodes can also act as transshipment nodes, we impose constraints (V6-A) and (V6-B).

$$t_i + s_i \leq 1, \forall i \in I \quad (\text{V6-A})$$

$$t_i + r_i \leq 1, \forall i \in I \quad (\text{V6-B})$$

$$t_i = \begin{cases} 1, & \text{if activated as a transshipment.} \\ 0, & \text{otherwise.} \end{cases}$$

2.2.7 Flow through a transshipment inequalities

When a node distributes, or receives any flow of CO₂ without being selected as a source or reservoir, then the t variable for this node must be selected as 1. Constraints (V8-A) and (V8-B) specify the relationship between flow and t variables.

$$\sum_{j \in N_i^+} x_{ij} \leq \begin{cases} \tau t_i, \forall i \in T \\ \tau(t_i + s_i), \forall i \in S \\ \tau(t_i + r_i), \forall i \in R \end{cases} \quad (\text{V8-A})$$

$$\sum_{j \in N_i^-} x_{ji} \leq \begin{cases} \tau t_i, \forall i \in T \\ \tau(t_i + s_i), \forall i \in S \\ \tau(t_i + r_i), \forall i \in R \end{cases} \quad (\text{V8-B})$$

In this case, any outgoing flow from node i must lead to activation of t_i or s_i , for any node in sources (S). This behavior is analogous for reservoir (R) nodes and pure transshipment nodes (T). Similarly, any incoming flow to node i must lead to activation of t_i or s_i , for any node in sources(S). Again, this behavior is similar if node i belongs to R or T. We use the target amount of CO₂ to be sequestered as a big M value for the upper-bound for the sum of all flows originating (for V8-A) or ending in a node (V8-B).

These inequalities are similar in spirit to (5) and (6) in *SimCCS*, where source and reservoir selection variables are activated by CO₂ captured and stored respectively. However, t_i here is not associated with any cost in the objective function. Also, (V6-A) and (V6-B) would prevent s_i and t_i , and r_i and t_i to be both one at the same time.

2.2.8 Pipes through a transshipment inequalities

In addition to (V8-A) and (V8-B), constraints (V9-A) and (V9-B) may strengthen the relationship between pipe selection and transshipment variables. In this case, (V9-A) ensures activation of t variable if there is any outgoing pipe from a node while (V9-B) does the same for an incoming pipe, provided that such node is not selected as a reservoir or source.

$$y_{ijd} \leq \begin{cases} t_i, \forall i \in T, j \in N_i^+, d \in D \\ (t_i + s_i), \forall i \in S, j \in N_i^+, d \in D \\ (t_i + r_i), \forall i \in R, j \in N_i^+, d \in D \end{cases} \quad (\text{V9-A})$$

$$y_{ijd} \leq \begin{cases} t_j, \forall j \in T, i \in N_j^-, d \in D \\ (t_j + s_j), \forall j \in S, i \in N_j^-, d \in D \\ (t_j + r_j), \forall j \in R, i \in N_j^-, d \in D \end{cases} \quad (\text{V9-B})$$

To see that (V9-A) and (V9-B) do not dominate each other, consider a fractional solution where $t_i = 0, \forall i \in S$. Then from (V8-A) we have $\sum_{j \in N_i^+} x_{ij} \leq \tau s_i$, which implies that

$s_i \geq \sum_{j \in N_i^+} x_{ij} / \tau$. From (V9-A) we have that $y_{ijd} \leq s_i$. Thus, it ensures that:

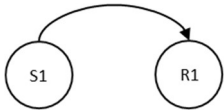
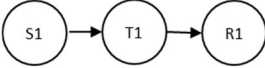
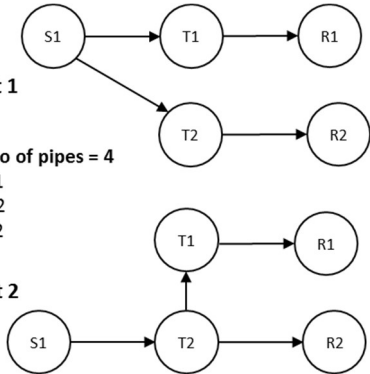
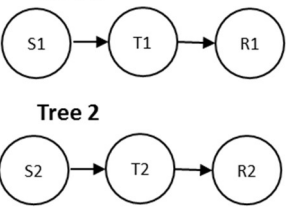
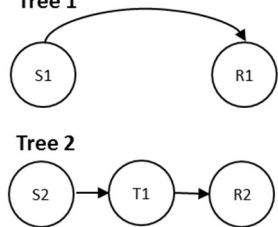
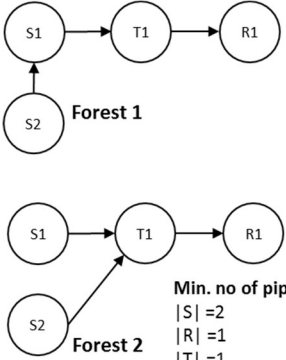
$s_i \geq \max\{ \sum_{j \in N_i^+} x_{ij} / \tau, y_{ijd} \}$, which is stronger than having (V8-A) and (V9-A)

individually. This analysis is similar for any t_i , where $i \in R$.

2.2.9 Inequality for minimum number of pipes (forest inequalities)

Using transshipment variables, we explore the network topology requirement for a given number of sources, reservoirs and transshipment nodes. Table 1 presents this analysis for the relation between the minimum required pipes and number of different types of nodes in *SimCCS*. Every cell in this matrix is an arrangement of such minimum number of arcs for a given combination of number of sources $|S|$, reservoirs $|R|$, and transshipment nodes $|T|$, in a solution. The network arrangement in a solution is a spanning forest, where a forest may contain one or multiple network trees. A solution of CCS could have different trees which are not connected to each other.

Table 1. Network analysis

<p>a.</p>  <p>Min. no of pipes = 1 $S = 1$ $R = 1$ $T = 0$</p>	<p>b.</p>  <p>Min. no of pipes = 2 $S = 1$ $R = 1$ $T = 1$</p>	<p>c.</p>  <p>Min. no of pipes = 4 $S = 1$ $R = 2$ $T = 2$</p>
<p>d.</p>  <p>Min. no of pipes = 4 $S = 2$ $R = 2$ $T = 2$</p>	<p>e.</p>  <p>Min. no of pipes = 3 $S = 2$ $R = 2$ $T = 1$</p>	<p>f.</p>  <p>Min. no of pipes = 3 $S = 2$ $R = 1$ $T = 1$</p>

For example, case (d) and (e). Cases (c) and (f) present 2 different solution forests each for the same minimum number of pipes within them. From this analysis, it can be inferred that the total minimum number of pipes on arcs is equal to $\text{Max}(|R|, |S|) + |T|$.

$$\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} \geq \sum_{i \in I} s_i + \sum_{i \in I} t_i \quad (V7-A)$$

$$\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} \geq \sum_{i \in I} r_i + \sum_{i \in I} t_i \quad (V7-B)$$

Constraints V7-A and V7-B force the pipe selection binaries to sum up higher than the minimum number of pipes. Note that the $\text{Max}(|R|, |S|)$ is formulated by two constraints such that the one with higher RHS value dominates the other. One of them will be redundant, thus maintaining the inference of this analysis. Figure 8 presents the benefit of these forest constraints. For *SimCCS* instance from Figure 4. (a), the LP relaxation solution including constraints (V6) & (V7) increases the optimal objective value from \$655.59 to \$726.30. In this case, LHS in (V7) is: $\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} = 0.218$, which is less than the RHS = 3 for (V7-A), RHS=0.62 for (V7-B). LHS for the solution in Figure 6 is $\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} = 3$.

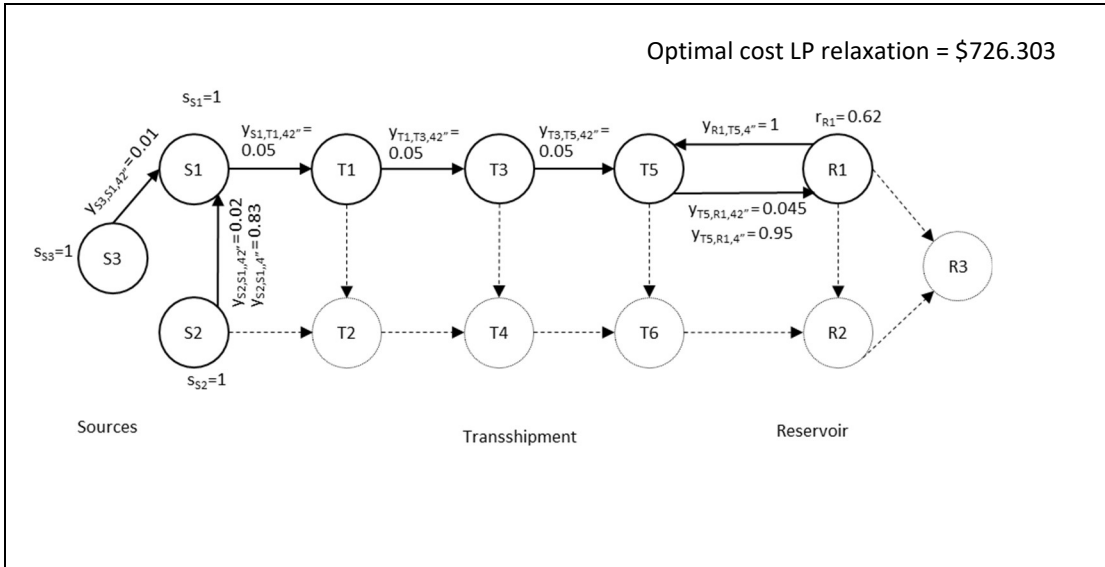


Figure 8. Example representing benefits of adding (V6), (V7)

2.2.10 Combined effect of valid inequalities

The effect of adding all the proposed valid inequalities changes the LP relaxation to at least the best of the LP relaxation obtained with individual constraints. Our empirical results suggest that after adding (V1) - (V9), there is further scope for improvement. The following valid inequalities (V10) and (V11) improve the linear relaxation solution of *SimCCS* with (V1) – (V9).

2.2.10.1 Transshipment requisite inequalities

Activation of any transshipment variable in the problem with all constraints ((V1), (V2), (V3) ... (V9)) involves transfer of CO₂ through many nodes, including those being selected as a source or a reservoir. A transshipment node cannot be activated without having at least one outgoing pipe (we enforce this by (V10-A)) or at least one incoming pipe (we enforce this by (V10-B)). These constraints impact the LP relaxation of the model with constraints (V1) – (V9).

$$t_i \leq \sum_d \sum_{j \in N_i^+} y_{ijd} \quad , \forall i \in I \quad (V10-A)$$

$$t_i \leq \sum_d \sum_{j \in N_i^-} y_{jid} \quad , \forall i \in I \quad (V10-B)$$

Figure 9 illustrates the improvement in the optimal LP relaxation of solution from \$936.69 to \$938.45 by adding (V10). Because the t variable for node ‘T3’ has the value of 1 in Figure 9. (a), then LHS = 1, which is greater than RHS for (V10-A) = $\sum_d \sum_{j \in N_i^+} y_{ijd} = 0.574$. Thus, the solution is infeasible for constraint (V10-A) for this node. This leads to an improvement in linear relaxation by building an additional pipe of 4 inch out of ‘T3’ of fractional value 0.379, which ensures binaries on RHS sum up to 1. Similar changes happen at node ‘R1’ where the t variable value is a fraction and there is a fractional pipe built out of it.

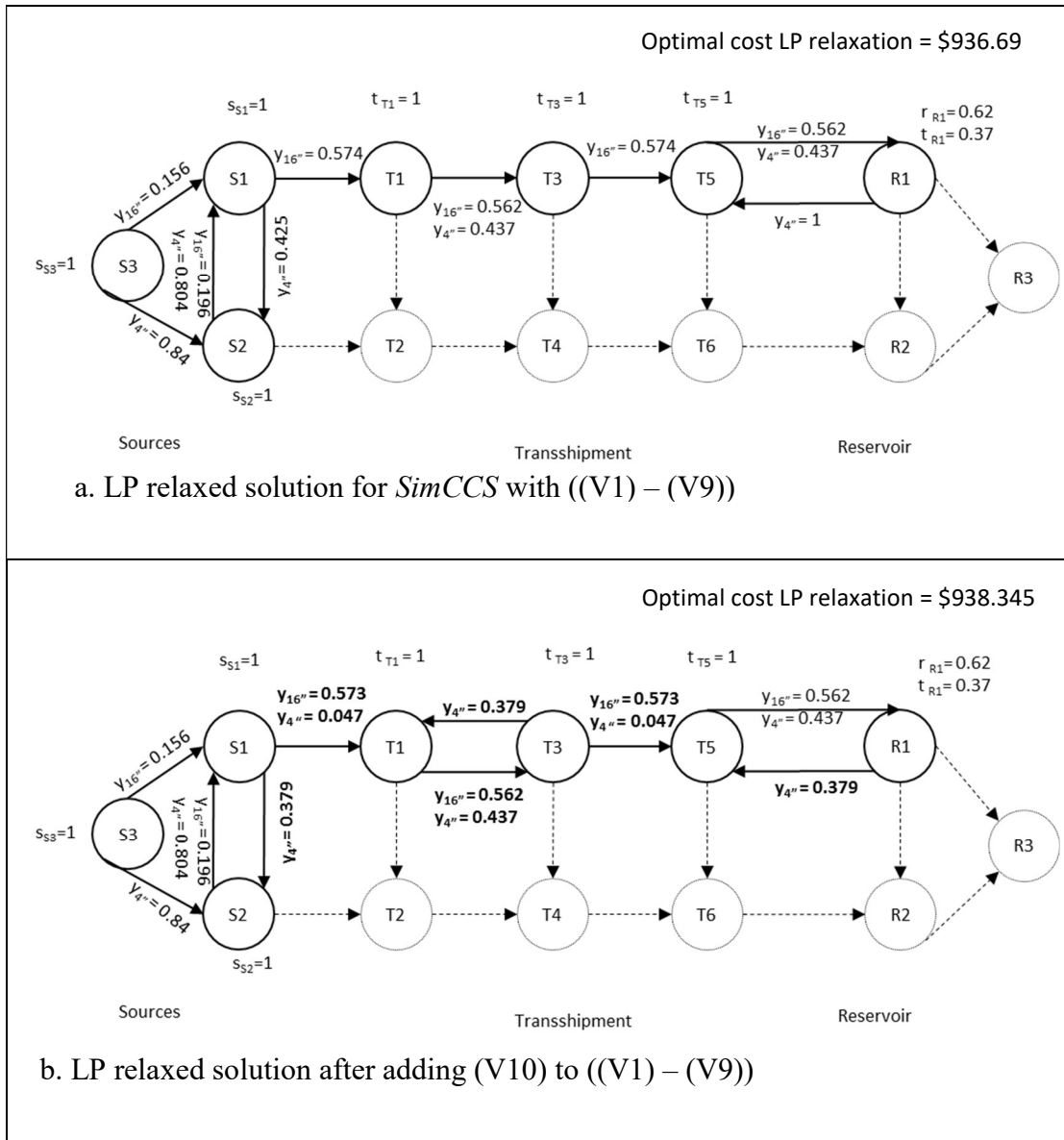


Figure 1. Example representing benefits of adding (V10) to combination of all constraints

2.2.10.2 Inequality to avoid looping pipes

Examination of previous improvements reveal that fractional pipes are built to and from a pair of nodes. To avoid this solution, constraint (V11) specifies that the total sum of all binaries for pipe selection between a pair of nodes must be less than 1. This is a consolidation of the constraint (8) from *SimCCS* where the direction of pipes was implied because an optimal solution would not build looping pipes but would rather send flow in one direction to minimize the cost.

$$\sum_{d \in D} Y_{ijd} + \sum_{d \in D} Y_{jid} \leq 1 \quad \forall (i, j) \in A, \text{ such that } i \in I \text{ and } j \in N_i^+ \quad (\text{V11})$$

Figure 10 illustrates the improvement in the *SimCCS* LP relaxation solution from Figure 9. (a) which increases from \$936.69 to \$940.63 by adding (V11). It clearly avoids back and forth pipes between the pairs ‘S1’ - ‘S2’ and ‘T5 - R1’ of Figure 9. (a), where $LHS = \sum_{d \in D} Y_{ijd} + \sum_{d \in D} Y_{jid} = 1.425$ and $= 2$ respectively. For both pairs, $LHS > RHS$ and hence the solution is infeasible for constraint (V11).

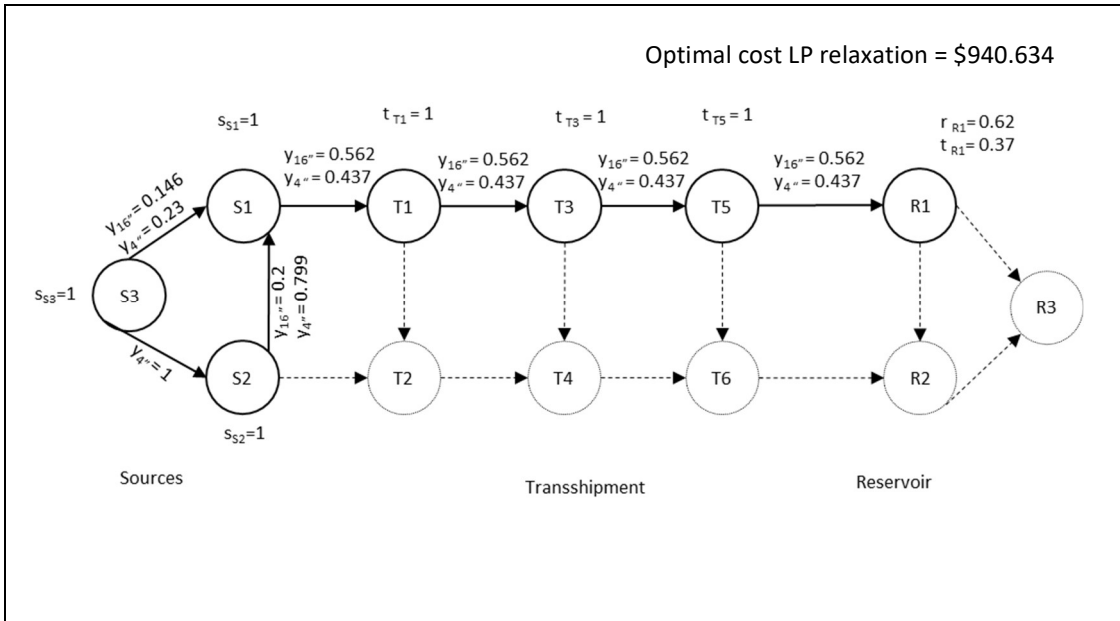


Figure 2. Example representing benefits of adding V11 to combination of constraints ((V1) – (V9))

3.0 COMPUTATIONAL EXPERIMENTS & RESULTS

The goal of this chapter is to present an analysis of the effectiveness of the introduced constraints and their effect on real applications and randomly generated instances. We refer to the combination of the *SimCCS* model and all introduced constraints from (V1) to (V11) as MIP'. The individual impact of adding each VI and activation constraints is provided along with the mathematical formulation. In the analysis of redundancy, we provide the effect of elimination of each constraint on the LP relaxation of MIP'. The aim of adding VIs is to reduce the optimal-solution time. We establish the performance of the VIs in terms of computation time, linear relaxations, and solution bounds of real problem and on randomly generated networks.

3.1 Analysis of effectiveness of valid inequalities.

This analysis validates all the VI constraints in the MIP' model to avoid redundancy. We conjectured that all VI's are essential and studied the changes in the LP relaxation for removal of each. For validation of effectiveness of a VI, its exclusion from MIP' must worsen the LP relaxation for at least one instance of the *SimCCS*. The justification of using each VI is presented in Table 2 for four instances with different configurations of $|N|$ nodes, $|A|$ arcs, $|S|$ sources, $|R|$ reservoirs and $|D|$ diameters. If the linear relaxation of MIP' worsened upon exclusion of a VI, a tick-mark (\checkmark) was entered in the corresponding cell. An x-mark (\times) was entered for the cell when the same linear relaxation of MIP' was obtained upon elimination of that VI.

Constraints (V8), (V10) and (V11) have tick-marks in every column revealing their significance in MIP' for these instances. Elimination of (V9) affects the network with 39 nodes and 252 arcs. This complements our previous discussion of including both (V8) and (V9) to activate the t variables, with neither dominating the other. Since constraints (V6) specify the distinction of a node being either a transshipment or a source/reservoir and are needed for the forest constraints to function, they are not included in this elimination test for validation. Since every row has at least one tick mark, the inclusion of each VI benefits MIP'.

Table 2. Test for validity of VIs on MIP'

Instance N , A , S , R , D	12,34,3,3,10	12,22,3,1,10	78,920,20,20,10	39,252,10,10,10	
Linear relaxation of MIP	655.23	361.29	8485.4	398.56	
Linear relaxation of MIP'	817.16	487.16	9152.56	572.04	
Linear relaxation upon elimination of VI from MIP'	V1	x	✓ 476.9	x	x
	V2	✓ 817.04	x	✓ 8943.99	✓ 571.24
	V3	x	x	✓ 8946.81	x
	V4	x	x	✓ 9151.92	x
	V5	x	x	✓ 9151.20	x
	V7	x	✓ 473.89	x	✓ 566.07
	V8	✓ 749.31	✓ 451.34	✓ 9111.98	✓ 441.24
	V9	x	x	x	✓ 571.98
	V10	✓ 816.09	✓ 396.81	✓ 9111.98	✓ 566.78
	V11	✓ 813.23	✓ 402.31	✓ 9151.97	✓ 567.03
Optimal solution	1131.04	497.95	*10489.4	861.9	

✓: LP relaxation worsens on removal of VI'

x: LP relaxation is same as MIP'

*: Solution after 3600 seconds on MIP'

3.2 Testing MIP' on a real problem

The aim of developing VI's is to solve large scale network problem sets of SIM CCS quickly and efficiently. Hence, we tested MIP' on a CCS problem for Alberta oil sands in Canada. This region with significant carbon emissions is a potential area for application of CCS. It consists of designing a CCS infrastructure to transport CO₂ from a set of oil-sands to regions of acid gas sites [22]. Figure 11 presents a map of Alberta and Figure 12 shows the spatial arrangement of the Alberta network with sources, reservoirs and potential arcs shown in red circles, blue circles, and grey lines, respectively. This network dataset has 22 sources and 16 reservoirs for CO₂ capture and storage, and 200 potential arcs and 10 available diameters of pipes. The 34 transshipment nodes and their corresponding arcs are not displayed in this figure. The maximum flow through this network based on various capacities of source, sinks and pipes is 39.1 megatons of CO₂ per year.

This problem was tested for different targets on Gurobi 7.0.2 solver working on a Core(TM) i7-5500 CPU@2.4 GHz consisting of 2 cores and 4 logical processors with 8 GB of RAM. For a target capture of 90 % of the maximum flow of CO₂, it took 31379 seconds for MIP to obtain the optimal solution of \$6874.73 million while MIP' could solve for the same optimal cost in 6,620 seconds. This presents a speedup of about five times. The optimal solution of \$687.95 million for a target of 10% of the maximum flow was solved by MIP in 12 seconds whereas MIP' took 2.5 seconds to solve it. For 50 % of the maximum flow, MIP solved for an optimal cost of \$3481.31 in 400 seconds while

MIP' solved it in 55 seconds. In general, MIP' solves this problem faster than MIP because of the improved linear relaxations and tighter feasible region.



[By NormanEinstein (Own work) [Public domain], via Wikimedia Commons]

Figure 11. Map of oil sands in Alberta

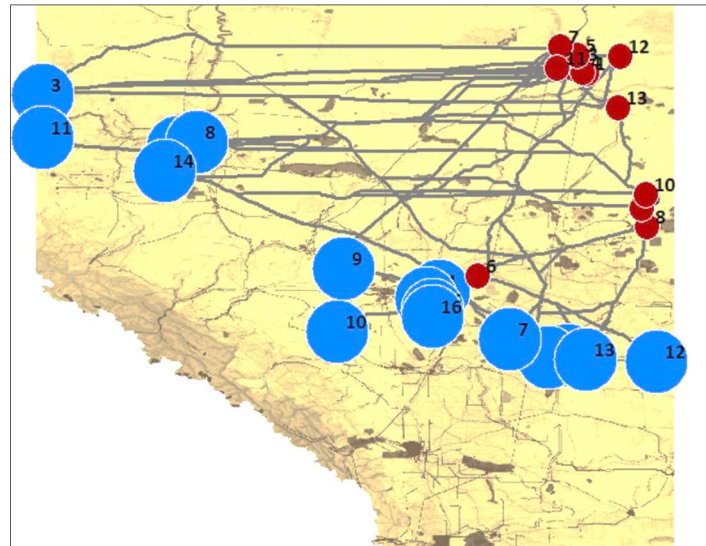


Figure 12. Alberta network for *SimCCS*, screenshot of the map from *SimCCS* integrated application.

3.3 Testing MIP' on generated datasets

The promising results of MIP' on Alberta network established its benefits for that structure of data. However, the solution time speedup of MIP' could change based on the network configuration, density, cost structure and target. Hence, we test our developments on randomly generated instances of a *SimCCS* network. The purpose of these experiments on various networks is to test the limits of using MIP'.

The main factors for this test were network size, density, number of diameters, and percentage of maximum flow of CO₂. We used a network grid of nodes with different configurations to generate instances for testing MIP'. A grid is defined by the number of sources, reservoirs and transshipments, assumed to be disjoint sets. Nodes are arranged from left to right as shown in Figure 13. Sources are connected to each other by arcs and the west most layer of transshipment nodes is connected to the adjacent sources.

Correspondingly, arcs connect all reservoirs to each other and the east most layer of transshipment nodes is connected to the adjacent reservoirs. For sparse networks, arcs connect each transshipment node to the existing node east, north, west or south of itself. For dense networks, along with the previous arcs, diagonal arcs connect each transshipment to any adjacent transshipment node. Figures 13 and 14 illustrate a generated sparse and dense network respectively with the stated rules for a 20-node instance consisting of 5 sources, 5 reservoirs and 10 transshipment nodes. The double arrows indicate arcs between a pair of nodes in both directions. We also designed grids of 3 other configurations of 39, 78 and 117 nodes.

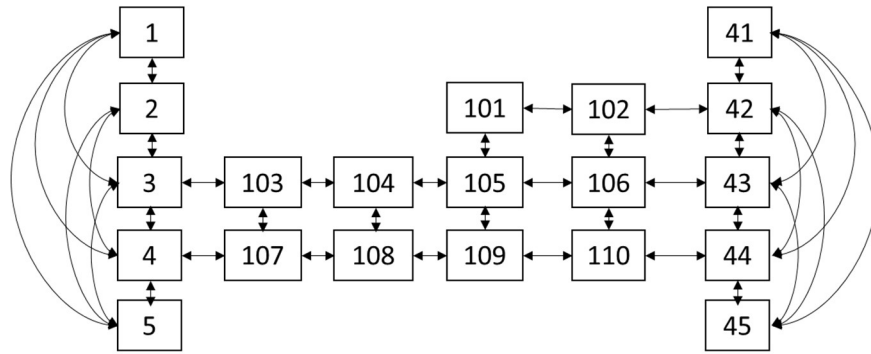


Figure 13. Sparse network of 5 nodes, 5 reservoirs and 10 transshipments

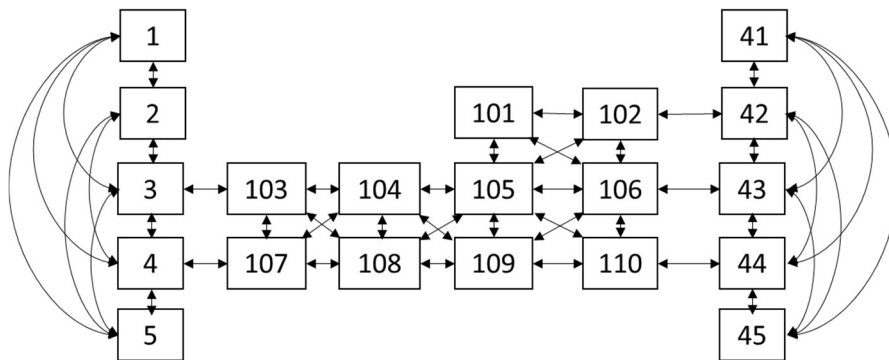


Figure 14. Dense network of 5 nodes, 5 reservoirs and 10 transshipments

We use 3 levels of pipe-diameter, consisting of 3, 7 and 10 diameters. We used the same pipe costs and capacities as in the Alberta problem data. The percentage of maximum flow had 3 levels - 10%, 50% and 90%. The costs and capacities for sources and reservoirs were randomly generated with a uniform distribution $U(a, b)$, where ‘a’ and ‘b’ represent the extreme limits of the range of costs derived from historical data. To test for data variability, we generated 4 replicates of datasets for each combination were generated using Python 2.7.12.

3.3.1 Computational results

We executed the maximum-flow LP problem for every network to calculate the maximum amount of CO₂ that can flow through it. The target τ value in the problem was then set to the percentage of maximum flow of the respective combination. We solved all instances on Gurobi 7.0.2 solver until optimality or time-limit of 1 hour, whichever occurred earlier. Tables 3.a and 3.b display the comparison of MIP and MIP' for each combination of parameters tested on the same machine used for the Alberta problem. Each row is one arrangement of network configuration, density, diameter and percentage of maximum flow for the CCS network. The time reported in each cell is the average of the 4 corresponding replicates. An 'x' mark indicates that all replicates timed out. The comparison parameter for speedup is calculated using the formula:

$$\text{Speedup} = \frac{\text{Solution time for MIP}}{\text{Solution time for MIP'}}$$

Speedup is calculated when at least one instance is solved for both MIP and MIP'. An average speedup greater than 1 indicates faster MIP' solution for an instance. We also report the percentage of instances solved to optimality before reaching time limit for every combination in Table 3. In terms of average speedup, MIP' is faster than MIP in 34 of the 47 combinations which solved to optimality within an hour. MIP' times out for all instances in sparse networks of 78 with 7 diameters, and 50% of the maximum flow. However, it solves a CCS problem on 78-node dense network with 10 possible pipes and 50% of the maximum flow in 413 seconds where MIP times out. All instances for 117-node sparse networks after 7 diameters start timing out for both MIP and MIP', so we did not test the dense networks.

Table 3. (a) Solution times of randomly generated instances – 20 and 39 nodes

Network config: S / R / T	Density A / N	Dia- meter D	% max- flow	MIP'(MIP + VI)		MIP		Average speedup (Speedup [instance]= MIP time/MIP' time)
				Average time (seconds)	% of instances solved to optimality	Average time (seconds)	% of instances solved to optimality	
5/ 5 /10	sparse (74/20)	3	10	0.12	100%	0.25	100%	2.2
			50	0.08	100%	0.07	100%	0.8
			90	0.20	100%	0.10	100%	0.5
		7	10	0.16	100%	1.62	100%	10.9
			50	0.72	100%	2.98	100%	4.3
			90	1.54	100%	4.10	100%	2.7
		10	10	0.14	100%	1.02	100%	6.9
			50	0.95	100%	8.47	100%	7.8
			90	2.86	100%	21.14	100%	10.5
	dense (92/20)	3	10	0.15	100%	0.99	100%	6.8
			50	0.09	100%	0.05	100%	0.6
			90	0.28	100%	0.24	100%	0.9
		7	10	0.30	100%	5.48	100%	21.4
			50	0.48	100%	3.13	100%	7.3
			90	3.90	100%	10.38	100%	2.4
		10	10	0.67	100%	6.35	100%	16.4
			50	0.65	100%	6.32	100%	16.0
			90	5.86	100%	37.66	100%	13.7
10/ 10 /19	sparse (252/39)	3	10	0.29	100%	2.31	100%	8.3
			50	0.60	100%	0.31	100%	0.5
			90	3.02	100%	0.64	100%	0.2
		7	10	1.32	100%	23.77	100%	24.0
			50	31.07	100%	46.13	100%	5.6
			90	148.45	100%	175.86	100%	2.8
		10	10	0.86	100%	15.19	100%	21.5
			50	51.08	100%	188.89	100%	11.9
			90	487.12	100%	545.67	100%	1.6
	dense (296/39)	3	10	0.24	100%	1.80	100%	7.5
			50	0.71	100%	0.30	100%	0.5
			90	6.60	100%	3.58	100%	0.8
		7	10	0.53	100%	48.41	100%	94.7
			50	117.76	100%	346.26	100%	2.4
			90	2621.95	100%	940.83	100%	0.4
		10	10	0.52	100%	45.29	100%	86.6
			50	43.49	100%	348.30	100%	25.2
			90	721.97	100%	1003.75	100%	3.6

Table 3. (b) Solution times of randomly generated instances -78 and 117 nodes

Network config: S / R / T	Density A / N	Dia- meter D	% max- flow	MIP'(MIP + VI)		MIP		Average speedup (Speedup [instance]= MIP time/MIP' time)
				Average time (seconds)	% of instances solved to optimality	Average time (seconds)	% of instances solved to optimality	
20/ 20 /38	sparse (920/78)	3	10	1.44	100%	3.70	100%	2.6
			50	718.41	100%	356.84	100%	0.4
			90	217.81	100%	185.69	100%	0.9
		7	10	8.45	100%	651.71	100%	79.5
			50	x	0%	2457.70	25%	-
			90	x	0%	x	0%	-
		10	10	12.05	100%	2439.34	100%	199.8
			50	x	0%	x	0%	-
			90	x	0%	x	0%	-
	dense (1022/78)	3	10	1.51	100%	3.08	100%	2.0
			50	x	0%	x	0%	-
			90	247.91	100%	1253.92	100%	6.9
		7	10	10.38	100%	1305.57	100%	173.3
			50	x	0%	x	0%	-
			90	x	0%	x	0%	-
		10	10	34.53	100%	1885.44	100%	105.5
			50	412.16	25%	x	0%	-
			90	x	0%	x	0%	-
30/ 30 /57	sparse (1984/117)	3	10	2088.34	25%	2531.92	75%	0.9
			50	x	0%	x	0%	-
			90	352.59	100%	216.05	100%	0.6
		7	10	66.41	25%	2123.89	25%	31.98
			50	x	0%	x	0%	-
			90	x	0%	x	0%	-
		10	10	x	0%	x	0%	-
			50	x	0%	x	0%	-
			90	x	0%	x	0%	-

In all 10- diameter instances, MIP' always beats MIP since all speed-ups are greater than 1. Additionally, MIP' loses to MIP in only 2 of the 21 combinations of 7 diameters. For those 3-diameter instances, MIP' solves slower than MIP in 11 of the 21 combinations. The advantage of MIP' over MIP is clear as the number of possible diameters increase. This is illustrated in the scatterplot in Figure 15.

In the instances with 10% of maximum flow of CO₂, speedup is higher than 1 for 18 of the 19 combinations which solve to optimality, the odd one being a dataset with 3 available diameters of pipes. For 50% and 90% targets, MIP' beats MIP consistently for arrangements of 7 and 10 diameters, but it has mixed results when only 3 pipe diameters are available. Figure 16 shows the effect of the maximum flow parameter with respect to speedups.

We compare the objective function values of MIP and MIP' for instances that timed out using an index calculated as: Total cost at timeout ratio = $\frac{\text{Objective function at timeout for MIP}}{\text{Objective function at timeout for MIP'}}$

This index compares the objective of the best integer solution found within an hour. The average of this ratio is reported in Table 4. A ratio greater than 1 indicates that MIP' provides a better solution than MIP when neither solved a problem to optimality. MIP' obtained a less expensive solution than MIP in 13 of the 17 combinations which did not solve to optimality. The results for 117-node networks are significant because the total cost at timeout ratio is greater than 1 for all its combinations. Although the speedup and total cost ratio depend on the instance structure as expected, the presented tests on randomly generated instances reveal a fairly successful impact of VI's on the *SimCCS* model.

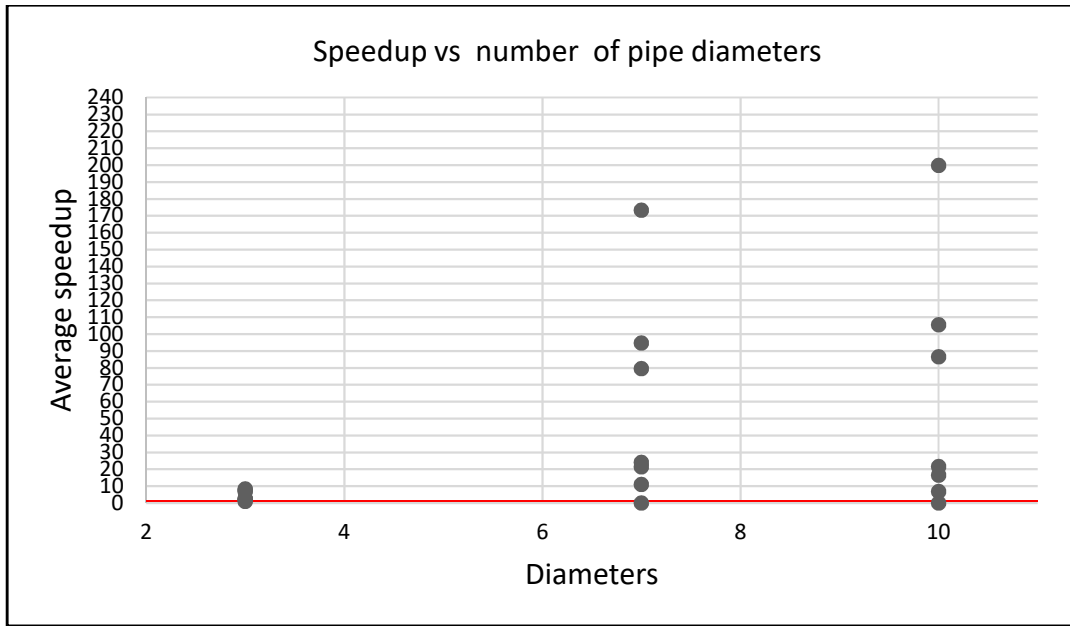


Figure 15. Scatter diagram of speed-up vs number of pipes

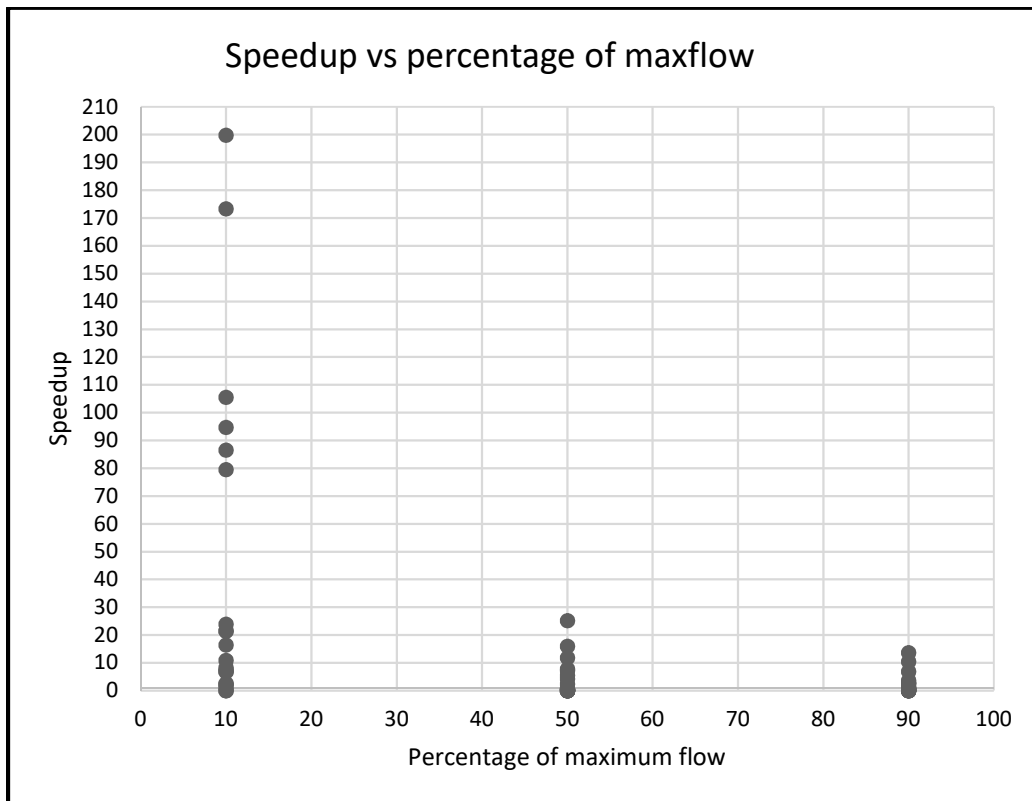


Figure 16. Scatter diagram of speed-up vs percentage of max-flow

Table 4. Objective function ratio of instances that time-out

Network config: S R T	Density A / N	Diameter D	% max-flow	Total cost at timeout ratio
20/ 20 /38	sparse (920/78)	3	50	1.000
		7	50	1.009
			90	1.000
		10	50	1.007
			90	1.002
		dense (1022/78)	7	50
	90			0.994
	10		50	0.999
			90	0.997
	30/ 30 /57	sparse (1984/117)	3	10
50				1.000
7			10	1.001
			50	1.005
			90	1.004
10			10	1.080
			50	1.103
			90	0.996

3.3.2 Consistent improvement in linear relaxation

In section 3.1 we validated each VI in MIP' by observing changes in the linear relaxations. Here, we test the magnitude of improvement in the linear relaxation for all combinations of the randomly generated networks. Tables 5. (a) and 5. (b) present the improvement of linear relaxation of MIP' over MIP when solved for the same instances reported in Table 3. The averaged comparison parameter is:

$$\text{Linear relaxation improvement index} = \frac{\text{Linear relaxation for MIP'}}{\text{Linear relaxation for MIP}}$$

An index higher than 1 signifies an improvement in linear relaxation for that combination of network. Tables 5. (a) and 5. (b) establish a minimum improvement of 6% of linear relaxation over MIP. MIP' obtained a better initial root node even in those instances where MIP was faster (13 of the 47 combinations). We visualize the proximity of the linear relaxation to its optimal solution in Figure 17.

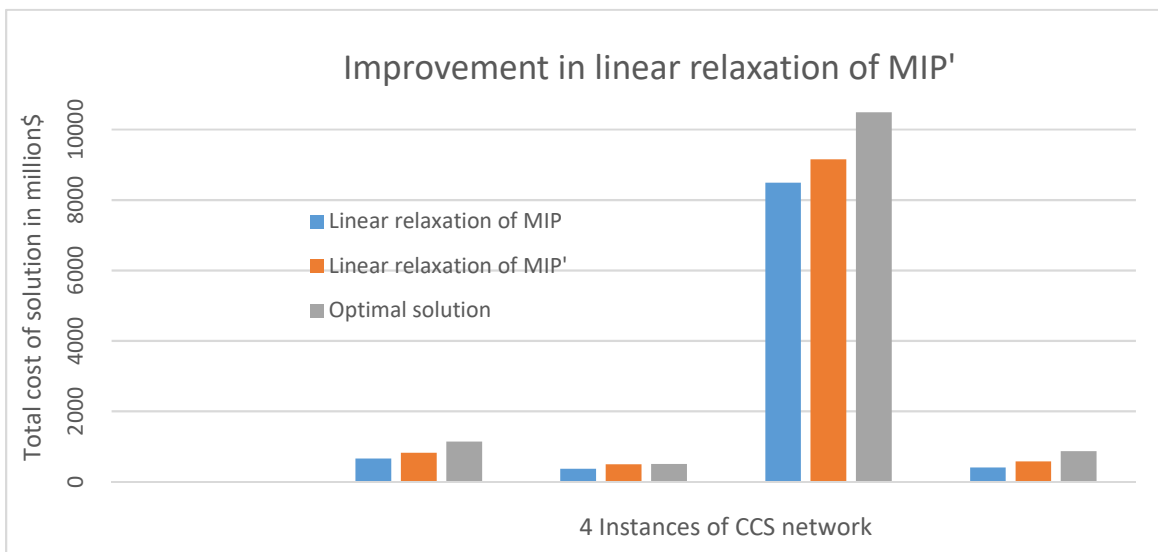


Figure 17. Proximity of linear relaxation to an optimal solution

Table 5. (a) Linear relaxation index of randomly generated instances – 20 & 39 nodes

Network config : S R T	Density A / N	Dia- meter D	% max-flow	Average speedup (Speedup [instance]= MIP time/MIP' time)	Linear relaxation improvement index (MIP' LP relax/ MIP LP relax)
5/ 5 /10	sparse (74/20)	3	10	2.2	2.55
			50	0.8	1.08
			90	0.5	1.06
		7	10	10.9	1.75
			50	4.3	1.11
			90	2.7	1.10
		10	10	6.9	1.97
			50	7.8	1.20
			90	10.5	1.13
	dense (92/20)	3	10	6.8	2.57
			50	0.6	1.07
			90	0.9	1.06
		7	10	21.4	1.86
			50	7.3	1.12
			90	2.4	1.07
		10	10	16.4	1.79
			50	16.0	1.18
			90	13.7	1.12
10/ 10 /19	sparse (252/39)	3	10	8.3	1.62
			50	0.5	1.07
			90	0.2	1.06
		7	10	24.0	1.36
			50	5.6	1.08
			90	2.8	1.07
		10	10	21.5	1.44
			50	11.9	1.12
			90	1.6	1.10
	dense (296/39)	3	10	7.5	1.64
			50	0.5	1.07
			90	0.8	1.06
		7	10	94.7	1.31
			50	2.4	1.06
			90	0.4	1.05
		10	10	86.6	1.43
			50	25.2	1.12
			90	3.6	1.10

Table 5. (b) Linear relaxation index of randomly generated instances – 78 & 117 nodes

Network config : S R T	Density A / N	Dia- meter D	% max-flow	Average speedup (Speedup [instance]= MIP time/MIP' time)	Linear relaxation improvement index (MIP' LP relax/ MIP LP relax)
20/ 20 /38	sparse (920/78)	3	10	2.6	1.12
			50	0.4	1.06
			90	0.9	1.05
		7	10	79.5	1.16
			50	-	1.05
			90	-	1.06
		10	10	199.8	1.22
			50	-	1.10
			90	-	1.09
	dense (1022/78)	3	10	2.0	1.11
			50	-	1.06
			90	6.9	1.06
		7	10	173.3	1.14
			50	-	1.06
			90	-	1.06
		10	10	105.5	1.22
			50	-	1.10
			90	-	1.08
30/ 30 /57	sparse (1984/117)	3	10	0.9	1.06
			50	-	1.06
			90	0.6	1.05
		7	10	-	1.11
			50	-	1.05
			90	-	1.06
		10	10	-	1.16
			50	-	1.09
			90	-	1.08

4.0 CONCLUSIONS AND FUTURE WORK

Carbon capture and storage is an important technological alternative for climate change mitigation [1,4]. The *SimCCS* model helps in determining the best network configuration in a CCS problem. However, the complexity of this variant of the FCNF problem, increases with the network size, making it difficult and long to solve. Commercial solvers are well equipped with various features but they often need to be coupled with innovative algorithms and constraints to solve such problems efficiently. In this work, we examined the structure of the problem and found a path of improvement. We identified problem-specific structures that solvers could not pick up, and that lead to the development of crucial valid inequalities to improve the *SimCCS* formulations. The analysis of redundancy of the VIs, results of a real large scale application, and computational experiments, established the significance of our improved formulation for *SimCCS*. Randomly generated datasets assist in testing the model for variations in structure of costs and capacities of capture, storage and transport. The wide range of network configurations help determine the scope of implementation of MIP'. As future work, we plan to solve variants of the *SimCCS* model, such as *SimCCS-Price*, to determine the effectiveness and benefits of our VIs.

After examining the *SimCCS* problem and its configuration, we identified the possible causes for long solution times to this problem. At first, we conjectured that the Benders decomposition approach would solve it efficiently. This involved decomposition of this problem into a network design master problem consisting of binary variables and a

network flow sub-problem with continuous variables. In this case, we expected a faster solution by decomposing the problem into two easier problems rather than solving one difficult integer problem. The modern Benders approach with the help of callbacks was integrated to test and solve CCS datasets. However, we observed a very slow convergence because of the problems evaluating many possible network alternatives. The faster decomposed problems accumulated a long time to solve because of many iterations. A large number of pipe-selection variables led to a difficult master problem with a large solution space. Moreover, we realized that more sophisticated techniques are needed to improve the flow of information between master and sub-problem. We realized that a modification of this approach would be needed to get competent results.

The future scope of optimization work on *SimCCS* would involve development of algorithms that avoid the shortcomings of the Benders approach. A method using decomposition of the problem based on typical solution structures could help solve large scale instances efficiently. An infinitesimal ratio of the number of pipes selected in a solution to the total available pipes could be a clue towards developing a method. An algorithm considering such sparse solution indices with selective decisions at iterations could be developed. Discovery of more valid inequalities based on assessment of complex solutions is another possibility. The benefits of VI's could be applied to similar other generalized FCNF problems involving pipeline design.

REFERENCES

- [1] Houghton, J.T., Meira Filho, L.G., Callender, B.A., Harris, N., Kattenberg, A. and Maskell, K., 1996. Climate change 1995: the science of climate change. Contribution of working group I to the second assessment of the intergovernmental panel on climate change.
- [2] Manne, A.S. and Richels, R.G., 1991. Global CO₂ emission reductions: the impacts of rising energy costs. In *Global Climate Change* (pp. 211-239). Springer Netherlands.
- [3] Schuiling, R.D. and de Boer, P.L., 2013. Six commercially viable ways to remove CO₂ from the atmosphere and/or reduce CO₂ emissions. *Environmental Sciences Europe*, 25(1), p.35.
- [4] Global CCS Institute, 2016. *The Global Status of CCS: 2016. Summary Report*, Australia.
- [5] Metz, B., Davidson, O., de Coninck, H., Loos, M. and Meyer, L., 2005. *IPCC Special Report: Carbon Dioxide Capture and Storage, Summary for Policymakers. A Report of Working Group III of the IPCC and Technical Summary. A Report Accepted by Working Group III of the IPCC but not Approved in Detail.*
- [6] Middleton, R.S. and Bielicki, J.M., 2009. A scalable infrastructure model for carbon capture and storage: *SimCCS*. *Energy Policy*, 37(3), pp.1052-1060.
- [7] Kuby, M.J., Bielicki, J.M. and Middleton, R.S., 2011. Optimal spatial deployment of CO₂ capture and storage given a price on carbon. *International Regional Science Review*, 34(3), pp.285-305.
- [8] Rothfarb, B., Frank, H., Rosenbaum, D.M., Steiglitz, K. and Kleitman, D.J., 1970. Optimal design of offshore natural-gas pipeline systems. *Operations research*, 18(6), pp.992-1020.
- [9] Bhaskaran, S. and Salzborn, F.J., 1979. Optimal design of gas pipeline networks. *Journal of the Operational Research Society*, 30(12), pp.1047-1060.
- [10] Singh, T.P., Neagu, N., Quattrone, M. and Briet, P., 2014, March. A decomposition approach to solve large-scale network design problems in cylinder gas distribution. In *International Conference on Operations Research and Enterprise Systems* (pp. 265-284). Springer, Cham.
- [11] Svensson, R., Odenberger, M., Johnsson, F. and Strömberg, L., 2004. *Transportation Infrastructure for CCS-Experiences and expected development. Greenhouse Gas Control Technologies*, 2005.

- [12] Dooley, J.J., Dahowski, R.T., Davidson, C.L., Wise, M.A., Gupta, N., Kim, S.H. and Malone, E.L., 2006. Carbon Dioxide Capture and Geologic Storage: A core element of a Global Energy Technology Strategy to address climate change. PNNL, Richland, WA.
- [13] Kobos, P., Malczynski, L., Borns, D.J. and McPherson, B.J., 2007, May. The ‘string of pearls’: the integrated assessment cost and source-sink model. In Conference Proceedings of the 6th Annual Carbon Capture & Sequestration Conference.
- [14] Middleton, R.S., Kuby, M.J., Wei, R., Keating, G.N. and Pawar, R.J., 2012. A dynamic model for optimally phasing in CO₂ capture and storage infrastructure. *Environmental Modelling & Software*, 37, pp.193-205.
- [15] Middleton, R.S., 2013. A new optimization approach to energy network modeling: anthropogenic CO₂ capture coupled with enhanced oil recovery. *International Journal of Energy Research*, 37(14), pp.1794-1810.
- [16] Johnson, N. and Ogden, J., 2012. A spatially-explicit optimization model for long-term hydrogen pipeline planning. *International Journal of Hydrogen Energy*, 37(6), pp.5421-5433.
- [17] Phillips, B.R. and Middleton, R.S., 2012. SimWIND: A geospatial infrastructure model for optimizing wind power generation and transmission. *Energy Policy*, 43, pp.291-302.
- [18] Streiffert, D., Philbrick, R. and Ott, A., 2005, June. A mixed integer programming solution for market clearing and reliability analysis. In Power Engineering Society General Meeting, 2005. IEEE (pp. 2724-2731). IEEE.
- [19] Atamtürk, A., Gómez, A. and Küçükyavuz, S., 2016. Three-partition flow cover inequalities for constant capacity fixed-charge network flow problems. *Networks*, 67(4), pp.299-315.
- [20] Balakrishnan, A., Li, G. and Mirchandani, P., 2017. Optimal Network Design with End-to-End Service Requirements. *Operations Research*, 65(3), pp.729-750.
- [21] Sherali, H.D. and Driscoll, P.J., 2000. Evolution and state-of-the-art in integer programming. *Journal of Computational and Applied Mathematics*, 124(1), pp.319-340.
- [22] Middleton, R.S. and Brandt, A.R., 2013. Using infrastructure optimization to reduce greenhouse gas emissions from oil sands extraction and processing. *Environmental Science & Technology*, 47(3), pp.1735-1744.
- [23] Magnanti, T.L. and Wong, R.T., 1984. Network design and transportation planning: Models and algorithms. *Transportation Science*, 18(1), pp.1-55.

- [24] Ghamlouche, I., Crainic, T.G. and Gendreau, M., 2003. Cycle-based neighbourhoods for fixed-charge capacitated multicommodity network design. *Operations Research*, 51(4), pp.655-667.
- [25] Costa, A.M., 2005. A survey on benders decomposition applied to fixed-charge network design problems. *Computers & Operations Research*, 32(6), pp.1429-1450.
- [26] Crainic, T.G., Gendron, B. and Hernu, G., 2004. A slope scaling/Lagrangian perturbation heuristic with long-term memory for multicommodity capacitated fixed-charge network design. *Journal of Heuristics*, 10(5), pp.525-545.
- [27] Rodríguez-Martín, I. and Salazar-González, J.J., 2010. A local branching heuristic for the capacitated fixed-charge network design problem. *Computers & Operations Research*, 37(3), pp.575-581.
- [28] Kim, D. and Pardalos, P.M., 1999. A solution approach to the fixed charge network flow problem using a dynamic slope scaling procedure. *Operations Research Letters*, 24(4), pp.195-203.
- [29] Barr, R.S., Glover, F. and Klingman, D., 1981. A new optimization method for large scale fixed charge transportation problems. *Operations Research*, 29(3), pp.448-463.
- [30] Balakrishnan, A., Li, G. and Mirchandani, P., 2017. Optimal Network Design with End-to-End Service Requirements. *Operations Research*, 65(3), pp.729-750.
- [31] Chouman, M., Crainic, T.G. and Gendron, B., 2016. Commodity Representations and Cut-Set-Based Inequalities for Multicommodity Capacitated Fixed-Charge Network Design. *Transportation Science*, 51(2), pp. 650-667.
- [32] Agarwal, Y.K. and Aneja, Y.P., 2017. Fixed charge multicommodity network design using p-partition facets. *European Journal of Operational Research*, 258(1), pp.124-135.
- [33] Paraskevopoulos, D.C., Bektaş, T., Crainic, T.G. and Potts, C.N., 2016. A cycle-based evolutionary algorithm for the fixed-charge capacitated multi-commodity network design problem. *European Journal of Operational Research*, 253(2), pp.265-279.
- [34] Gendron, B., Hanafi, S. and Todosijević, R., 2016. An Efficient Metaheuristic for the Multicommodity Fixed-Charge Network Design Problem. *IFAC-PapersOnLine*, 49(12), pp.117-120.
- [35] Hewitt, M., Nemhauser, G.L. and Savelsbergh, M.W., 2010. Combining exact and heuristic approaches for the capacitated fixed-charge network flow problem. *INFORMS Journal on Computing*, 22(2), pp.314-325.
- [36] Ortega, F. and Wolsey, L.A., 2003. A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem. *Networks*, 41(3), pp.143-158.

- [37] Monteiro, M.S., Fontes, D.B. and Fontes, F.A., 2011, July. An ant colony optimization algorithm to solve the minimum cost network flow problem with concave cost functions. In Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation (pp. 139-146). ACM.
- [38] Raj, K.A.A.D. and Rajendran, C., 2012. A genetic algorithm for solving the fixed-charge transportation model: two-stage problem. Computers & Operations Research, 39(9), pp.2016-2032.
- [39] Padberg, M.W., Van Roy, T.J. and Wolsey, L.A., 1985. Valid linear inequalities for fixed charge problems. Operations Research, 33(4), pp.842-861.
- [40] Stallaert, J., 2000. Valid inequalities and separation for capacitated fixed charge flow problems. Discrete Applied Mathematics, 98(3), pp.265-274.
- [41] Gurobi Optimization, I., 2016. Gurobi Optimizer Reference Manual; 2015. URL <http://www.gurobi.com>.

APPENDIX A

MIP' CONSISTING OF CONSTRAINTS ((1) TO (14), (V1) TO (V11))

Minimize $\sum_{i \in S} (F_i^S s_i + V_i^S a_i) + \sum_{i \in I} \sum_{j \in N_i^+} \sum_d F_{ijd}^P y_{ijd}$

$$+ \sum_{i \in I} \sum_{j \in N_i^+} V_{ij}^P x_{ij} + \sum_{j \in R} (F_j^R r_j + V_j^R b_j) \quad (1)$$

$$x_{ij} - \sum_{d \in D} Q_{ijd}^{\text{pmax}} y_{ijd} \leq 0, \forall i \in I, j \in N_i^+ \quad (2)$$

$$x_{ij} - \sum_{d \in D} Q_{ijd}^{\text{pmin}} y_{ijd} \geq 0, \forall i \in I, j \in N_i^+ \quad (3)$$

$$\sum_{j \in N_i^+} x_{ij} - \sum_{j \in N_i^-} x_{ji} - a_i + b_i = 0, \forall i \in I \quad (4)$$

$$a_i - Q_i^S s_i \leq 0, \forall i \in S \quad (5)$$

$$b_j - Q_j^R r_j \leq 0, \forall j \in R \quad (6)$$

$$\sum_{i \in S} a_i \geq \tau \quad (7)$$

$$\sum_{d \in D} y_{ijd} \leq 1, \forall i \in I, j \in N_i^+ \quad (8)$$

$$y_{ijd} \in \{0,1\}, \forall i \in I, j \in N_i^+, d \in D \quad (9)$$

$$s_i \in \{0,1\}, \forall i \in S \quad (10)$$

$$r_j \in \{0,1\}, \forall j \in R \quad (11)$$

$$x_{ij} \geq 0, \forall i \in I, j \in N_i^+ \quad (12)$$

$$a_i \geq 0, \forall i \in S \quad (13)$$

$$b_j \geq 0, \forall j \in R \quad (14)$$

$$t_i \in \{0,1\}, \forall i \in I \quad (15)$$

$$\tau \leq \sum_{j \in R} Q_j^R r_j \quad (V1)$$

$$s_i \leq \sum_d \sum_{j \in N_i^+} y_{ijd} + r_i, \forall i \in I \quad (V2)$$

$$r_i \leq \sum_d \sum_{j \in N_i^-} y_{jid} + s_i, \forall i \in I \quad (V3)$$

$$y_{ikd} \leq \sum_{f \in D} \sum_{j \in N_k^+} y_{kjf} + r_k, i, k \in I: (i, k) \in A, d \in D: Q_{ikd}^{\text{pmin}} = 0 \quad (V4)$$

$$y_{kid} \leq \sum_{f \in D} \sum_{j \in N_k^-} y_{j kf} + s_k, \forall k, i \in I: (k, i) \in A, d \in D: Q_{ikd}^{\text{pmin}} = 0 \quad (V5)$$

$$t_i + s_i \leq 1, \forall i \in I \quad (\text{V6-A})$$

$$t_i + r_i \leq 1, \forall i \in I \quad (\text{V6-B})$$

$$\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} \geq \sum_{i \in I} s_i + \sum_{i \in I} t_i \quad (\text{V7-A})$$

$$\sum_{i \in I} \sum_{j \in N_i^+} \sum_{d \in D} y_{ijd} \geq \sum_{i \in I} r_i + \sum_{i \in I} t_i \quad (\text{V7-B})$$

$$\sum_{j \in N_i^+} x_{ij} \leq \begin{cases} \tau t_i, \forall i \in T \\ \tau(t_i + s_i), \forall i \in S \\ \tau(t_i + r_i), \forall i \in R \end{cases} \quad (\text{V8-A})$$

$$\sum_{j \in N_i^-} x_{ji} \leq \begin{cases} \tau t_i, \forall i \in T \\ \tau(t_i + s_i), \forall i \in S \\ \tau(t_i + r_i), \forall i \in R \end{cases} \quad (\text{V8-B})$$

$$y_{ijd} \leq \begin{cases} t_i, \forall i \in T, j \in N_i, d \in D \\ (t_i + s_i), \forall i \in S, j \in N_i, d \in D \\ (t_i + r_i), \forall i \in R, j \in N_i, d \in D \end{cases} \quad (\text{V9-A})$$

$$y_{ijd} \leq \begin{cases} t_j, \forall j \in T, i \in N_j^-, d \in D \\ (t_j + s_j), \forall j \in S, i \in N_j^-, d \in D \\ (t_j + r_j), \forall j \in R, i \in N_j^-, d \in D \end{cases} \quad (\text{V9-B})$$

$$t_i \leq \sum_d \sum_{j \in N_i^+} y_{ijd} \quad , \forall i \in I \quad (\text{V10-A})$$

$$t_i \leq \sum_d \sum_{j \in N_i^-} y_{jid} \quad , \forall i \in I \quad (\text{V10-B})$$

$$\sum_{d \in D} y_{ijd} + \sum_{d \in D} y_{jid} \leq 1 \quad \forall (i, j) \in A, \text{ such that } i \in I \text{ and } j \in N_i^+ \quad (\text{V11})$$