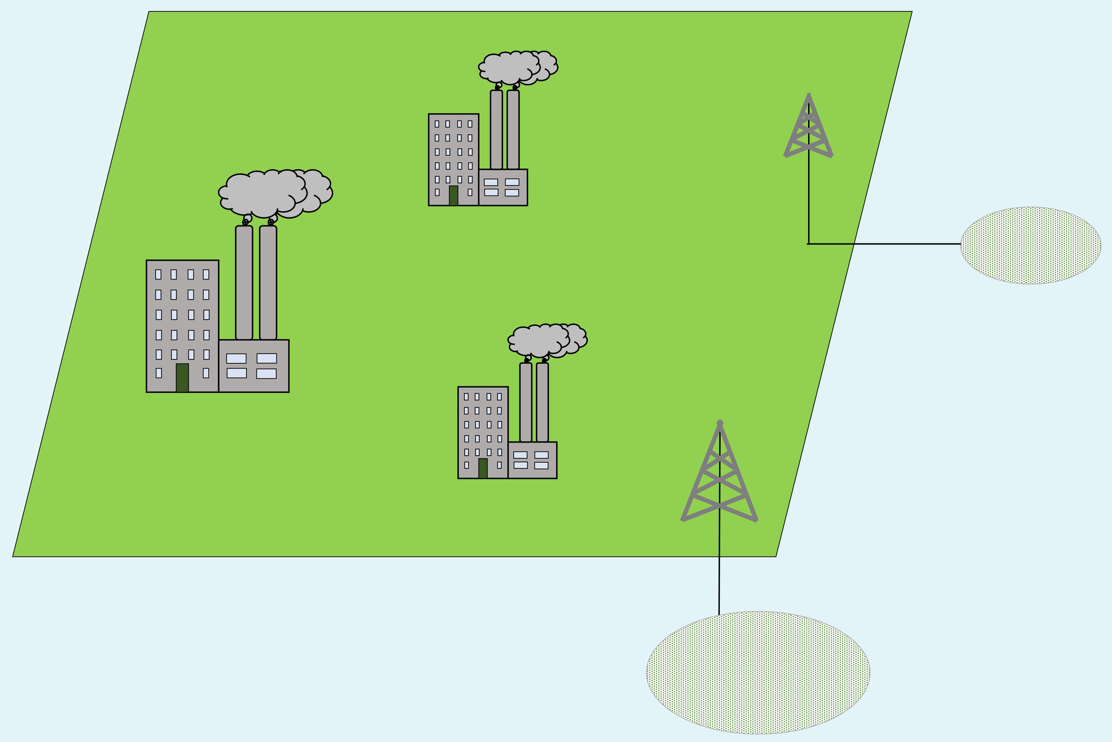
The Minimum Affine-Cost Flow Problem

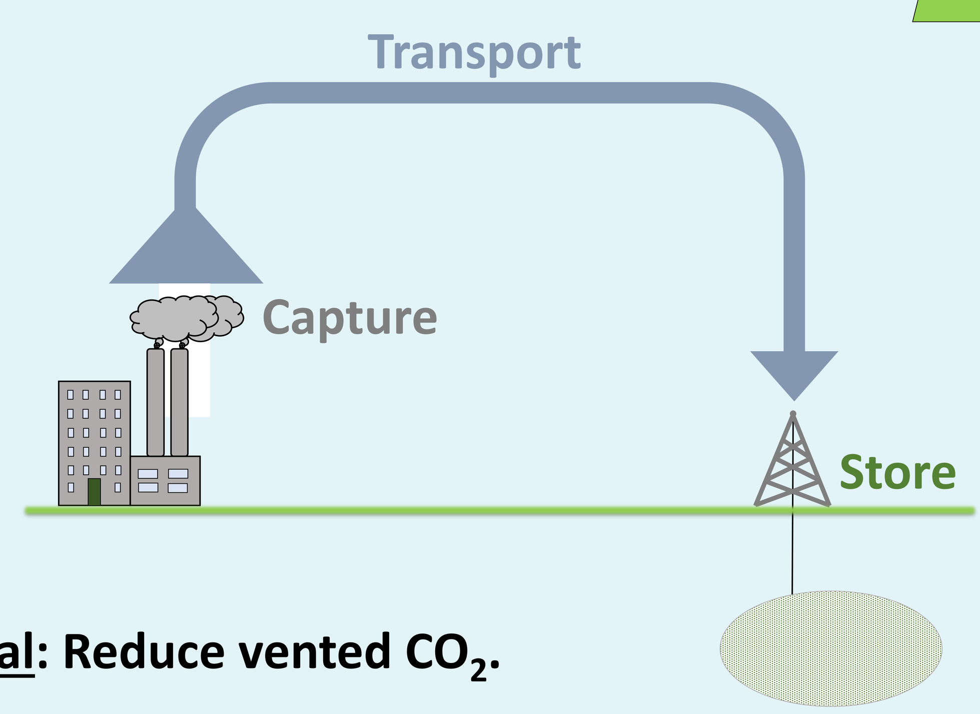
A Theoretical Approach to Designing Carbon Capture Networks

Kelsey Lally & Nathaniel Sauerberg | Mentors – Dr. Brendan Mumey & Dr. Sean Yaw | August 2019

Background Information

CO_2 capture and storage (CCS) is a climate change mitigation technology whose goal is to reduce the emission of greenhouse gases by capturing CO_2 emissions from industrial sources, transporting the CO_2 via a dedicated pipeline network, and storing it underground by injecting it into geological reservoirs.



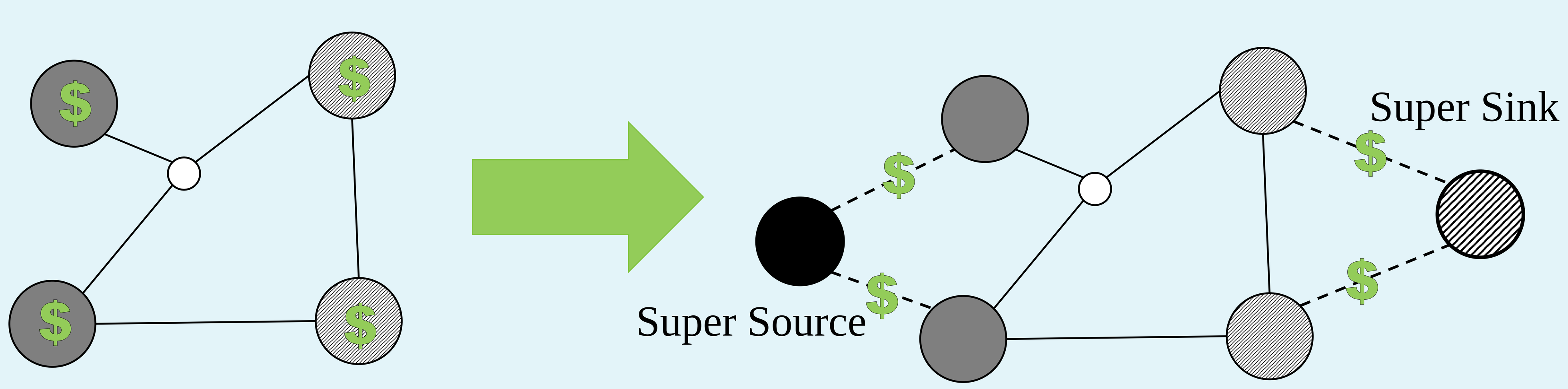


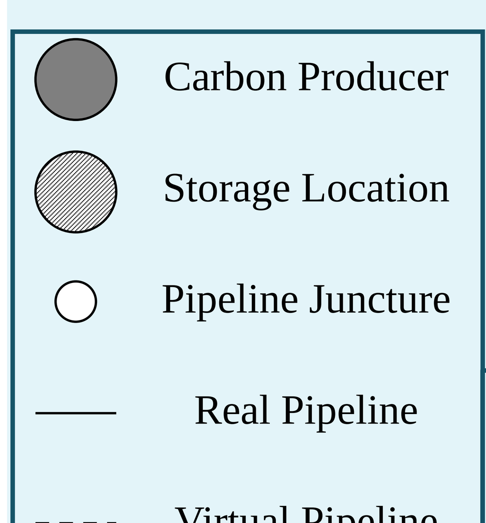
Goal: Reduce vented CO_2.

Designing CCS infrastructure is a large scale optimization problem. Specific sources, reservoirs, and pipeline routes need to be selected to capture a target amount of CO2. What sources should be used? Which reservoirs should be used? Where should pipelines be built to connect them?

Model Transformation

Original CCS Network **Associated Flow Network**





Under the MACF model, we generate an artificial super source and super sink. Costs and capacities for the original sources and sinks are moved onto virtual edges incident to the super source or sink. This allows our model to consider only one source and one sink, and to work exclusively with edge costs and capacities.

$ {
- Fixed cost to open facility/ pipeline
- Variable cost to extract/ store/ transport carbon (cost per ton)
- Carbon that can be captured/ stored/ transported (tons per year)

Integer Linear Program

This problem can be formulated as an ILP with the following input parameters and decision variables:

$G = (V, E)$	Undirected multi-graph
$s \in V$	Distinguished source
$t \in V$	Distinguished sink
$D \in \mathbb{R}^+$	Required amount of flow
$cap : E \to \mathbb{R}^+$	Capacity function to map edges to their capacities
$cost_f : E \to \mathbb{R}^+$	Fixed cost function to map edges to their fixed costs
$cost_v : E \to \mathbb{R}^+$	Variable cost function to map edges to their variable costs
$y_e \in \{0,1\}$	Indicates whether edge e is opened or not
$p_e \in \mathbb{Z}$	Gives the flow along edge e

The objective function aims to minimize the total cost of constructing edges between the super source and super sink to meet the demands

$$min \sum_{e \in E} y_e * cost_f(e) + p_e * cost_v(e)$$

Subject to the following constraints:

$f_{out}(s) = D$ (A)

$f_{in}(v) - f_{out}(v) = 0$ for all $v \in V \setminus \{s, t\}$ (B)

$y_e * cap(e) - p_e \geq 0$ for all $e \in E$ (C)

$p_e \geq 0$ for all $e \in E$ (D)

Where:

(A) Guarantees that the required amount of carbon is stored
(B) Flow is conserved at all internal nodes
(C) Flow along an edge never exceeds amount allowed by the edge's capacity and open/ closed status
(D) Flow along all edges is positive

Heuristic Algorithms

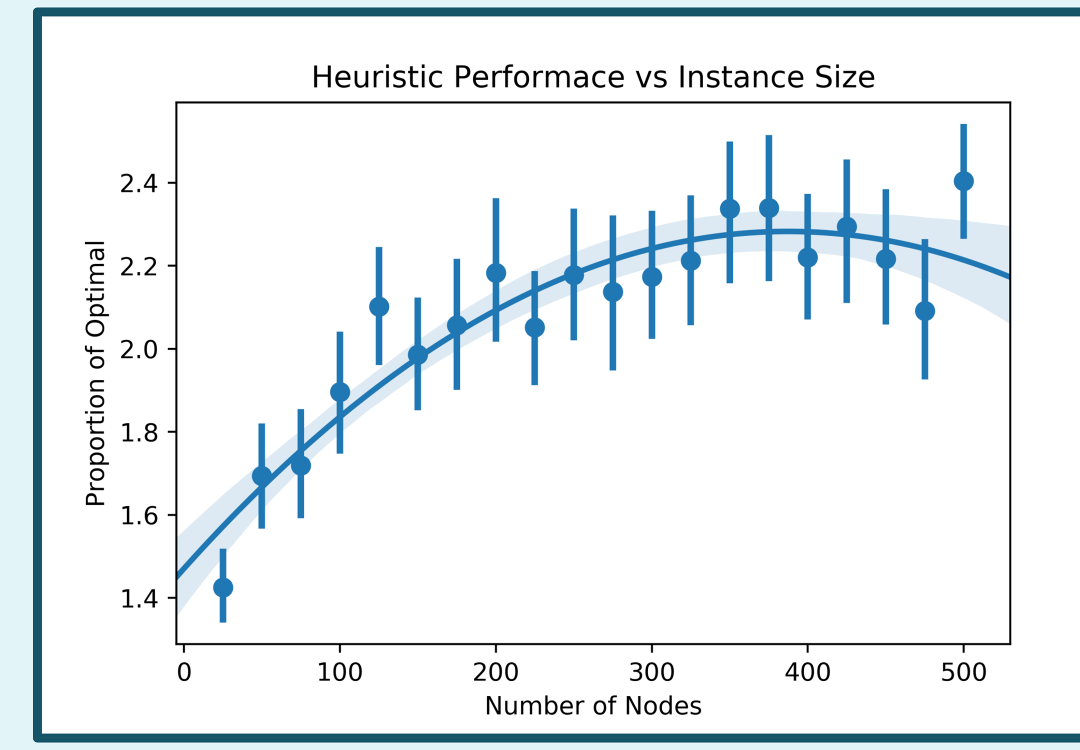
We designed three greedy algorithms for the Min-Cost Flow problem, all of which repeatedly saturate an s-t path until the demand is reached. The way this path is chosen distinguishes the algorithms. All three algorithms are easily computable, but none have a theoretical guarantee on the quality of the solution obtained.

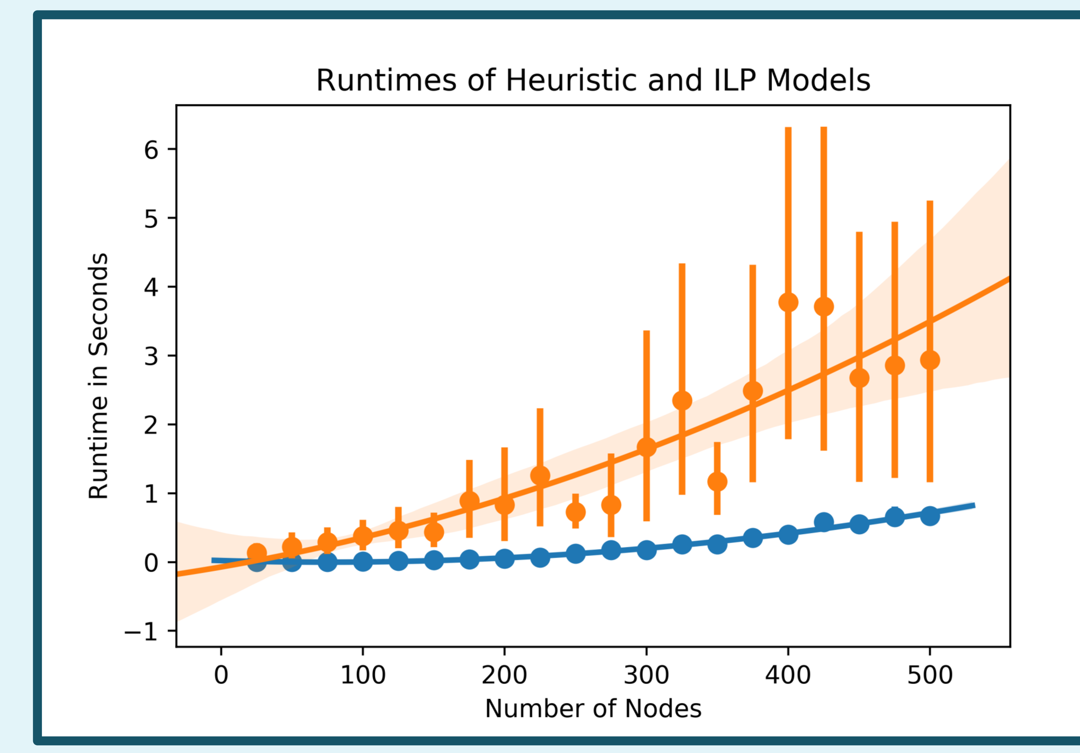
Heuristic 1: Select the s-t path for which the fixed cost to increase flow is minimal
Heuristic 2: Select the s-t path fully saturating a source or sink that has the best cost to flow ratio
Heuristic 3: Among the paths considered by the first two heuristics, select the one with the best cost to flow ratio

Results

Because the CCS problem is NP-hard to solve optimally, the exact ILP solution becomes infeasible on large inputs. Hence, an inexact but time-efficient solution, such as our heuristic algorithms, might be necessary to solve some real-world instances of the problem.

We implemented the ILP model and Heuristic 1 to compare their performance on randomly generated MACF instances.





We tested the ILP and Heuristic 1 on randomly generated graphs of 25 to 500 nodes. The upper graph shows the typical range of the optimality ratios that the heuristic was able to achieve, with the optimal solution being computed by the ILP. The lower graph shows how the runtimes of the two algorithms increase with the size of the input.

We plan to continue developing heuristic algorithms for CCS, and hope to develop an approximation algorithm that provides a theoretical guarantee on the solution quality.

References
[1] R. S. Middleton, M. J. Kuby, R. Wei, G. N. Keating, and R. J. Pawar, "A dynamic model for optimally phasing in CO2 capture and storage infrastructure," Environmental Modelling & Software, vol. 37, pp. 193–205, May 2012.

MONTANA STATE UNIVERSITY

NSF