CPLEX's Issues with our program

Shang-Lin Hsu

Yuning Yu

Things to discuss:

- CPLEX's algorithm while solving Mixed Integer Quadratic Program (MIQP)
- Problem we face: looping and long running time

Branch and cut algorithm:

CPLEX solves MIQP models in the same way as it solves other mixed integer programs (MIP).

Branch and cut involves running a **branch and bound algorithm** and using **cutting planes** to tighten the possible solutions.

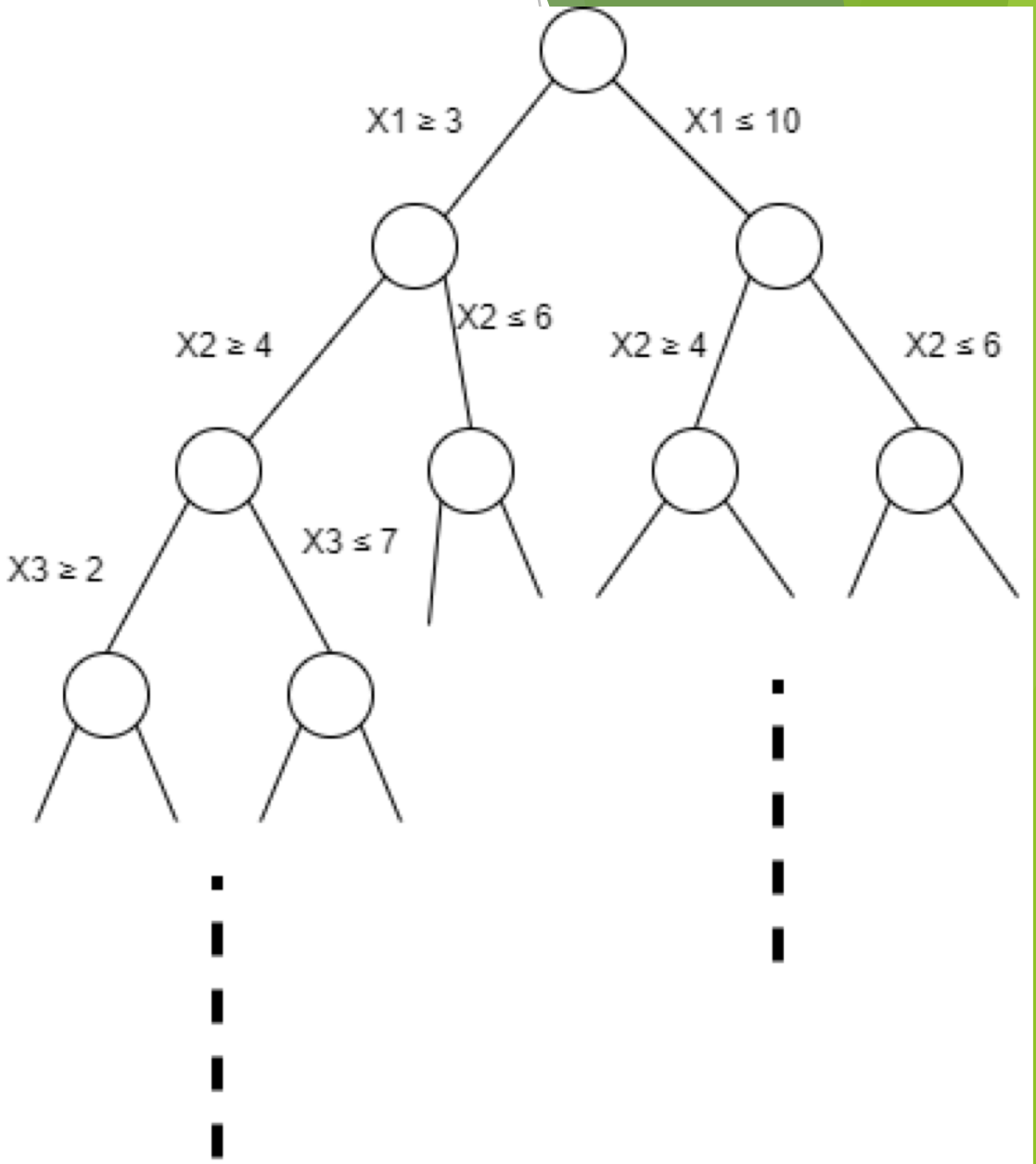
- **Branch and bound algorithm:** A tree that start from a root node and branched by the unknown variables' upper-bound and lower-bound. Thus, it is a balanced binary tree structure.

- **Cutting planes:** Cutting technique is used to reduce the nodes. If a child node's solution is greater then parent node's solution, the algorithm may stop diving deeper from this child node.

Branch and cut algorithm:

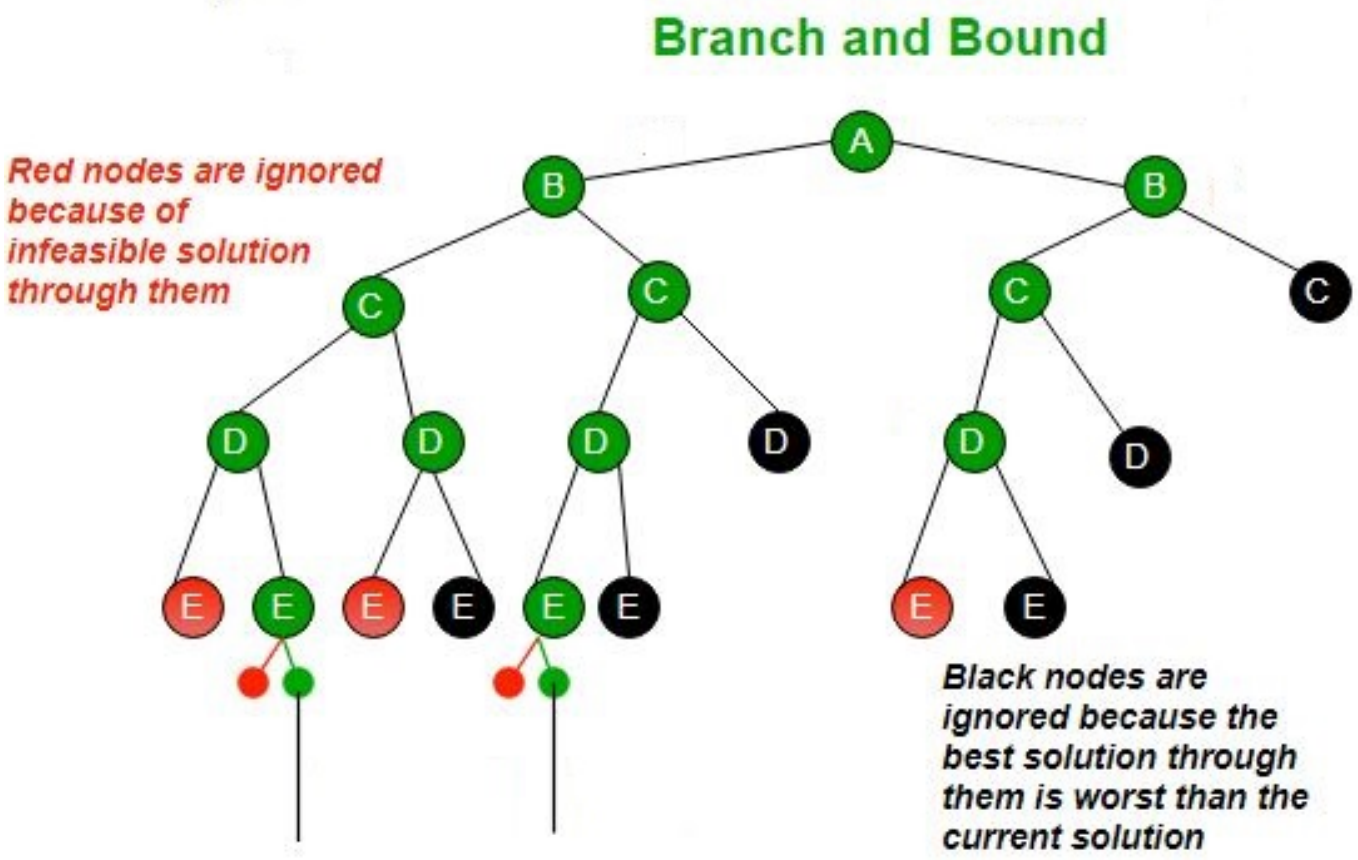
CPLEX will first calculate the lower-bound and the upper-bound form the model's constraints.

```
c100: - x100''12' + x12''100' - x100''13' + x13''100' - x100''23' + x23''100'
      - x100''24' + x24''100' - x100''25' + x25''100' - x100''31' + x31''100'
      - x100''34' + x34''100' - x100''36' + x36''100' - x100''38' + x38''100'
      - x100''42' + x42''100' - x100''43' + x43''100' - x100''44' + x44''100'
      - x100''54' + x54''100' - x100''65' + x65''100' - x100''68' + x68''100'
      - x100''88' + x88''100' - x100''90' + x90''100' - x100''91' + x91''100'
      - x100''97' + x97''100' - x100''99' + x99''100' - x100''101  = 0
c101: 0.4096 x64''1' + 0.4096 x34''1' + 0.4096 x66''1' + 0.4096 x99''1'
      + 0.4096 x4''1' + 0.4096 x36''1' + 0.4096 x68''1' + 0.4096 x70''1'
      + 0.4096 x8''1' + 0.4096 x41''1' + 0.4096 x46''1' + 0.4096 x15''1'
      + 0.4096 x81''1' + 0.4096 x51''1' + 0.4096 x83''1' + 0.4096 x54''1'
      + 0.4096 x25''1' + 0.4096 x89''1' + 0.4096 x60''1' + 0.4096 x61''1'
      + 0.4096 x31''1' + 0.4096 x95''1' + 21.9866 x1''64' + 4.6471 x1''34'
      + 4.1144 x1''66' + 8.1568 x1''99' + 21.3676 x1''4' + 19.9124 x1''36'
      + 0.7021 x1''68' + 5.7774 x1''70' + 19.0431 x1''8' + 5.0198 x1''41'
      + 8.8949 x1''46' + 6.298 x1''15' + 12.0228 x1''81' + 7.18 x1''51'
      + 23.1998 x1''83' + 16.3132 x1''54' + 23.7635 x1''25' + 4.3751 x1''89'
      + 9.7336 x1''60' + 22.1336 x1''61' + 0.6988 x1''31' + 12.4753 x1''95'
      <= 3500
```



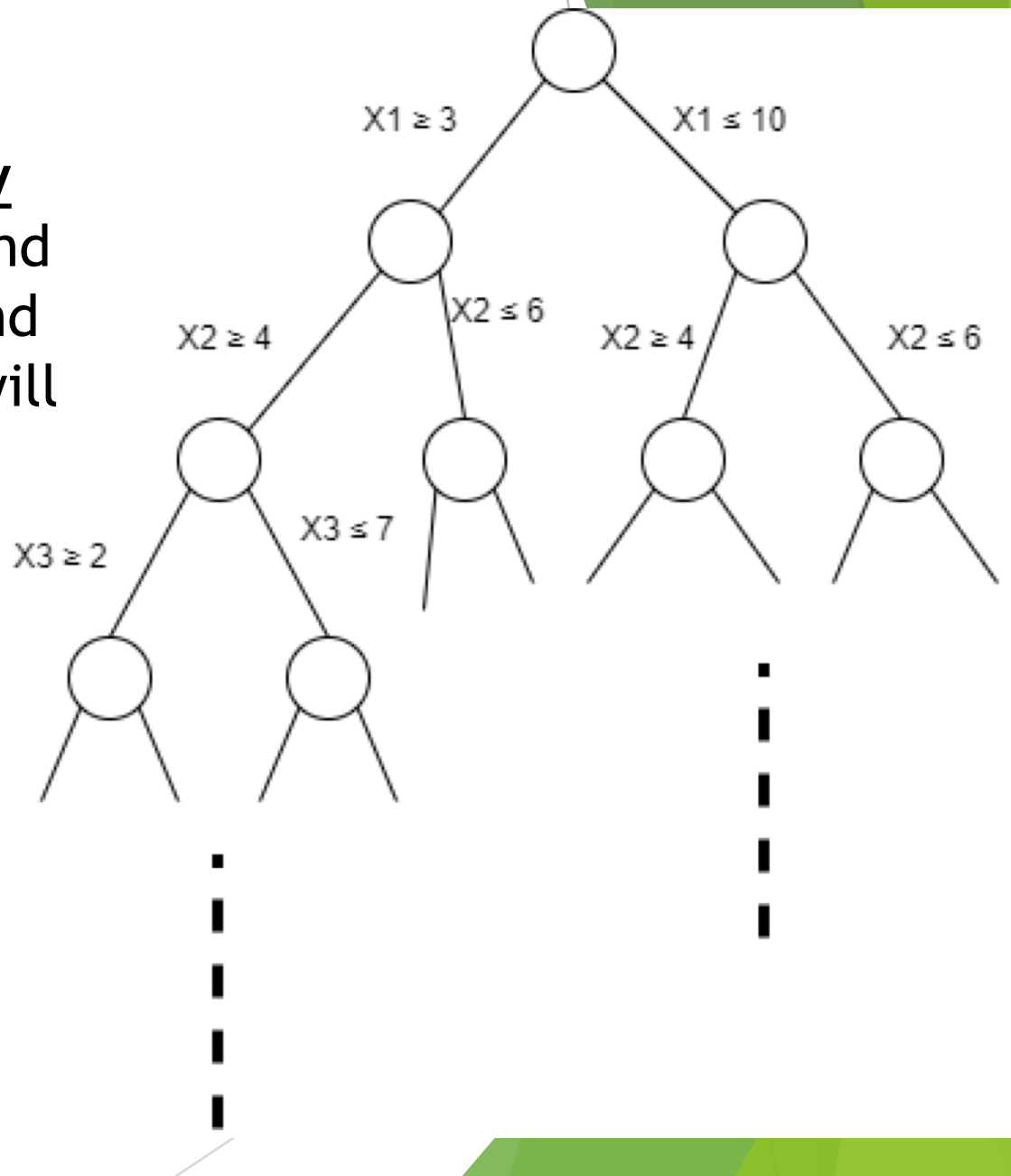
Branch and cut algorithm:

Cutting techniques is used to reduce the nodes.



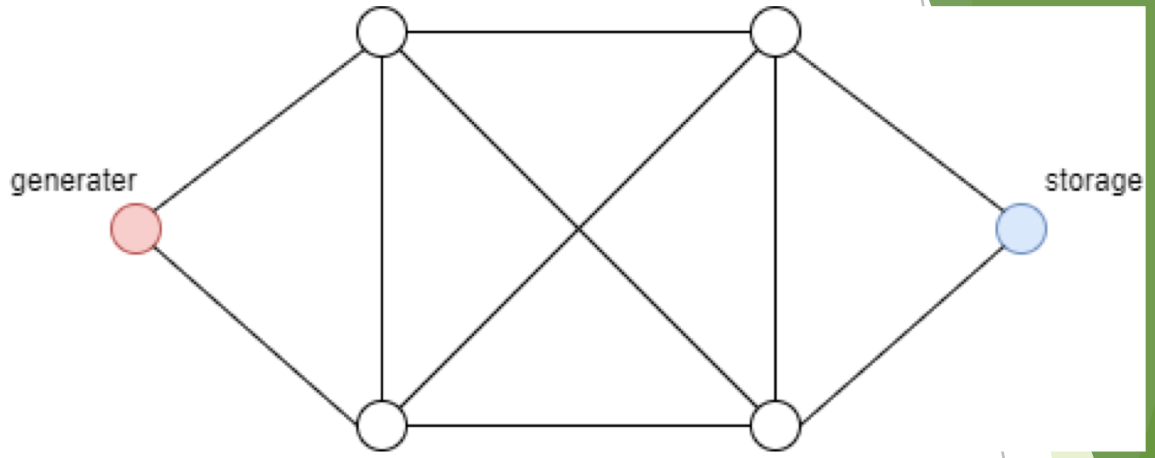
Looping problem :

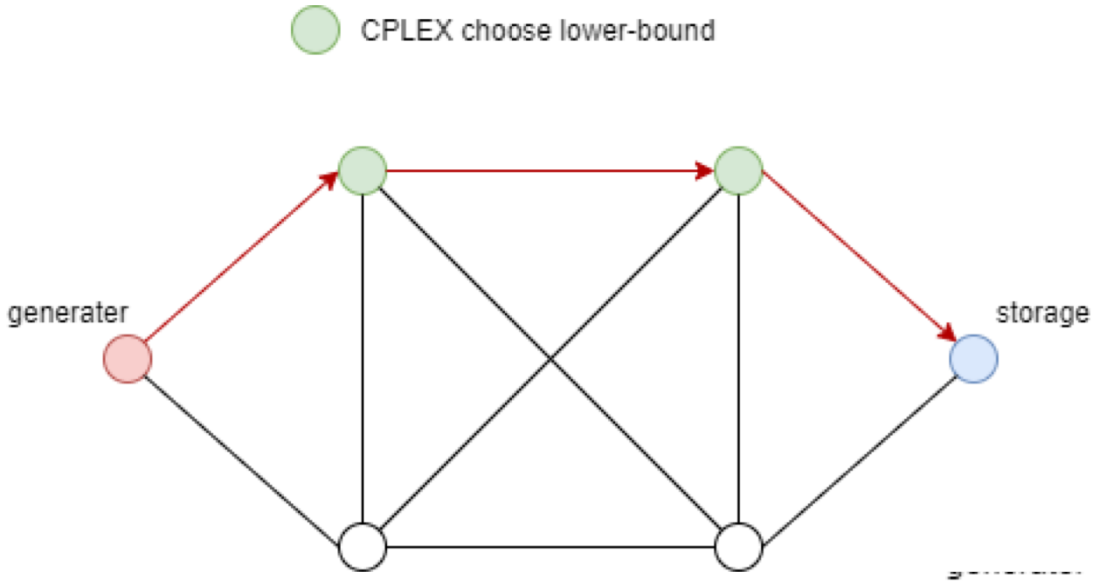
1. CPLEX will **<u>decide which way to go by itself</u>** (upper-bound first or lower-bound first). If CPLEX starts from upper-bound and return a optimal solution, there will have a loop problem.

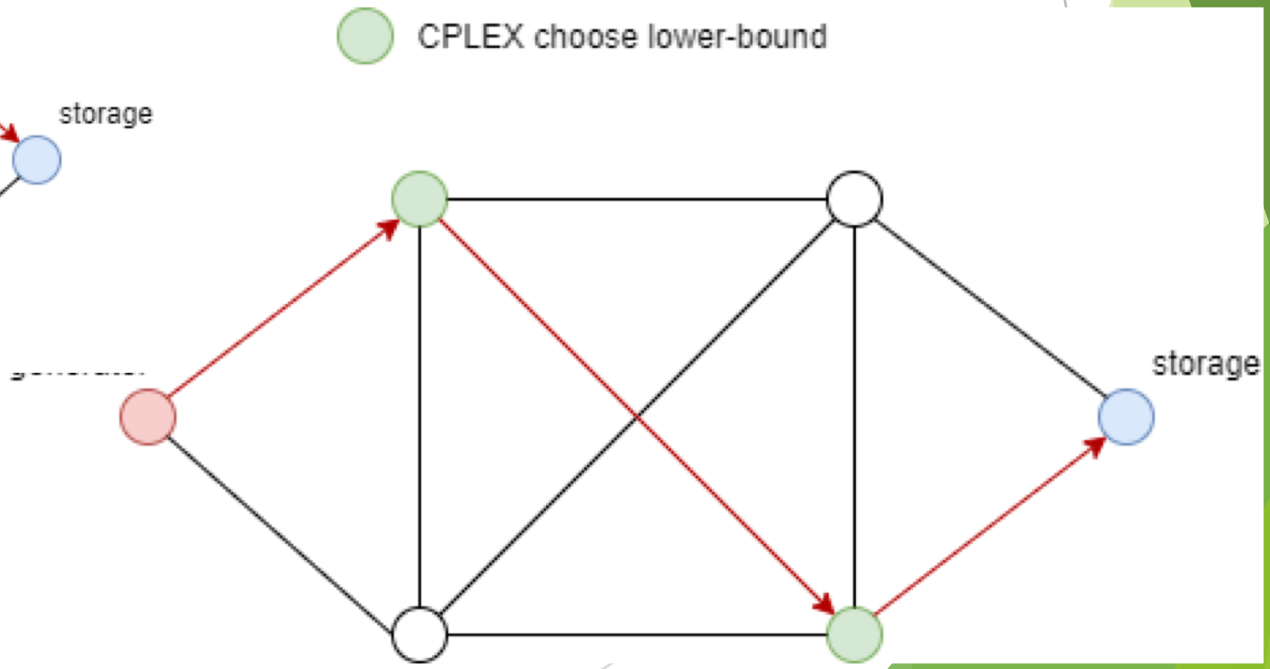


Looping problem :

Acceptable path:



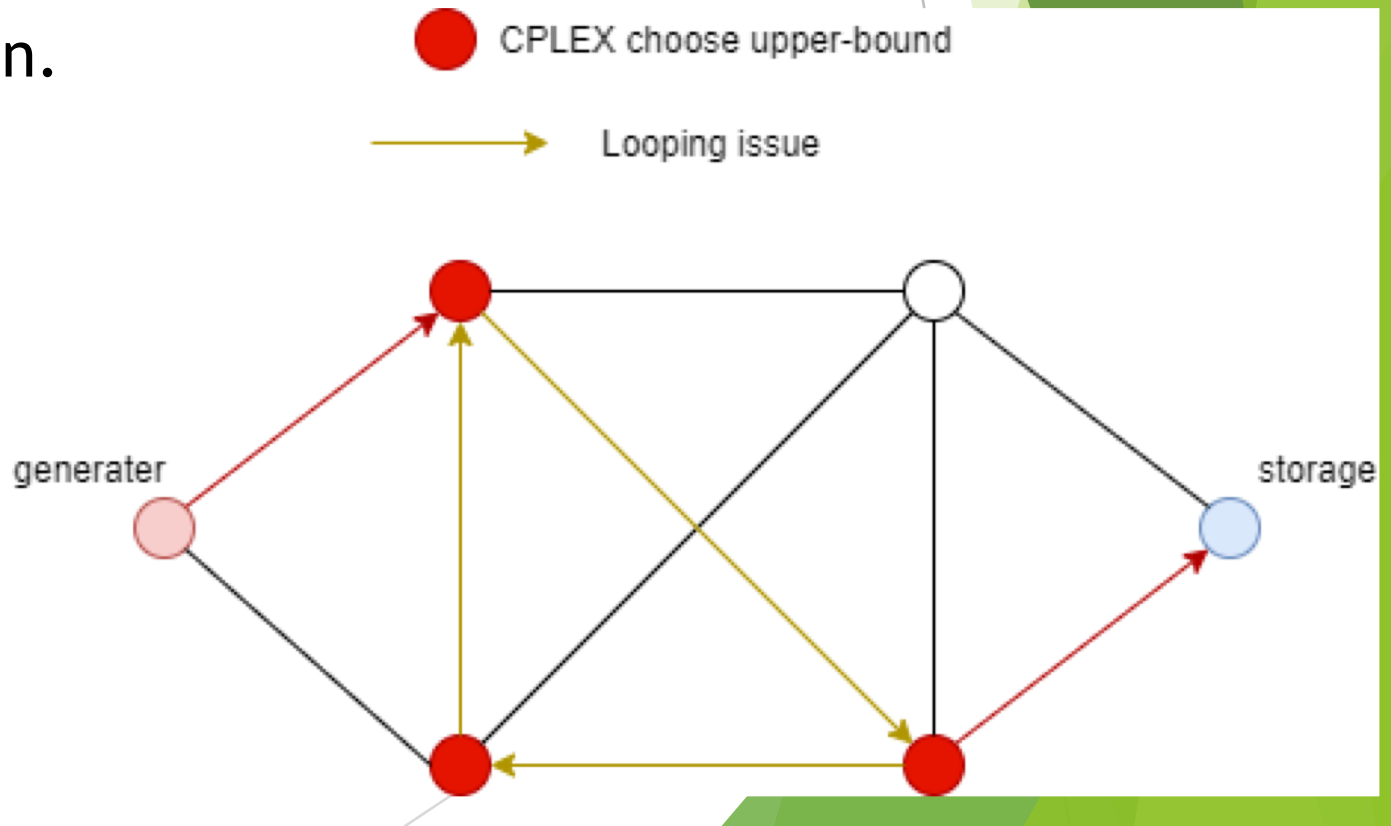




Looping problem :

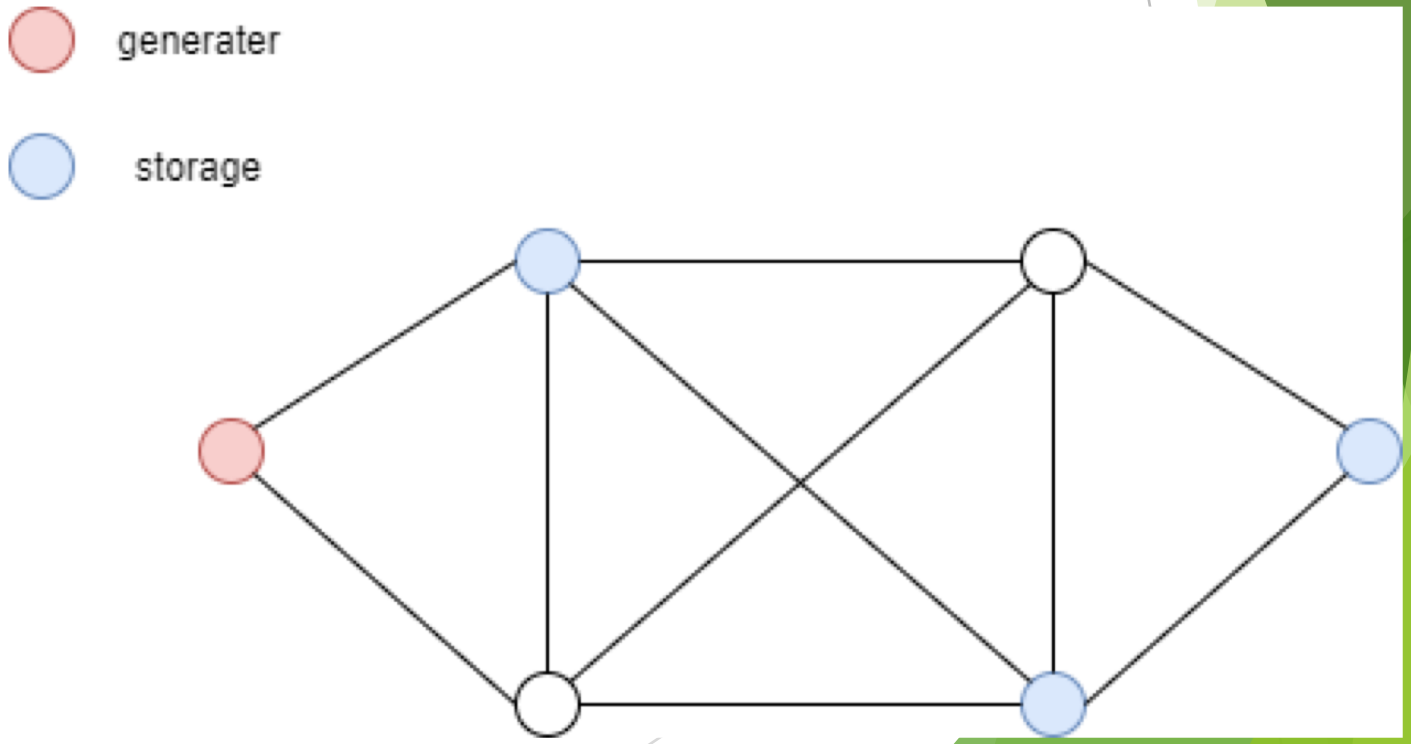
Bad path (CPLEX choose upper-bound):

This path **dose not influence** the optimal data resilience level, so CPLEX may return this path as a optimal solution.



Looping problem :

In some case, **more storage nodes are need to receive data**, so **no loop will happen**. (storage nodes have to minimize their energy cost to meet the optimize data resilience level)



Looping problem (solutions):

- We can manually calculate the result form the output file.

- We can force the algorithm to search from lower-bound first. Thus, the looping problem can be reduced. However, if we do this, the computation time and memory will increase significantly since the optimal solution may not be at the lower-bound branch.

Looping problem (Gap):

	Node	Nodes Left	Objective	IInf	Best Integer	Best Bound	Cuts/ ItCnt	Gap
	0	0	6605694.5215	0		6605694.5215	16	
*	0+	0			6111176.3346	6602366.6789		8.04%
	0	2	6602366.6789	0	6111176.3346	6602366.6789	16	8.04%
Elapsed time = 5.84 sec. (8847.15 ticks, tree = 0.02 MB, solutions = 1)								
	42	41	6601786.8747	0	6111176.3346	6602366.6789	523	8.04%
	79	78	6601395.0098	0	6111176.3346	6602366.6789	763	8.04%
	121	119	6597627.0738	0	6111176.3346	6602366.6789	1508	8.04%
	154	153	6593325.1586	0	6111176.3346	6602366.6789	2970	8.04%
	185	176	6583390.0191	0	6111176.3346	6602366.6789	6313	8.04%

Gap tolerance :

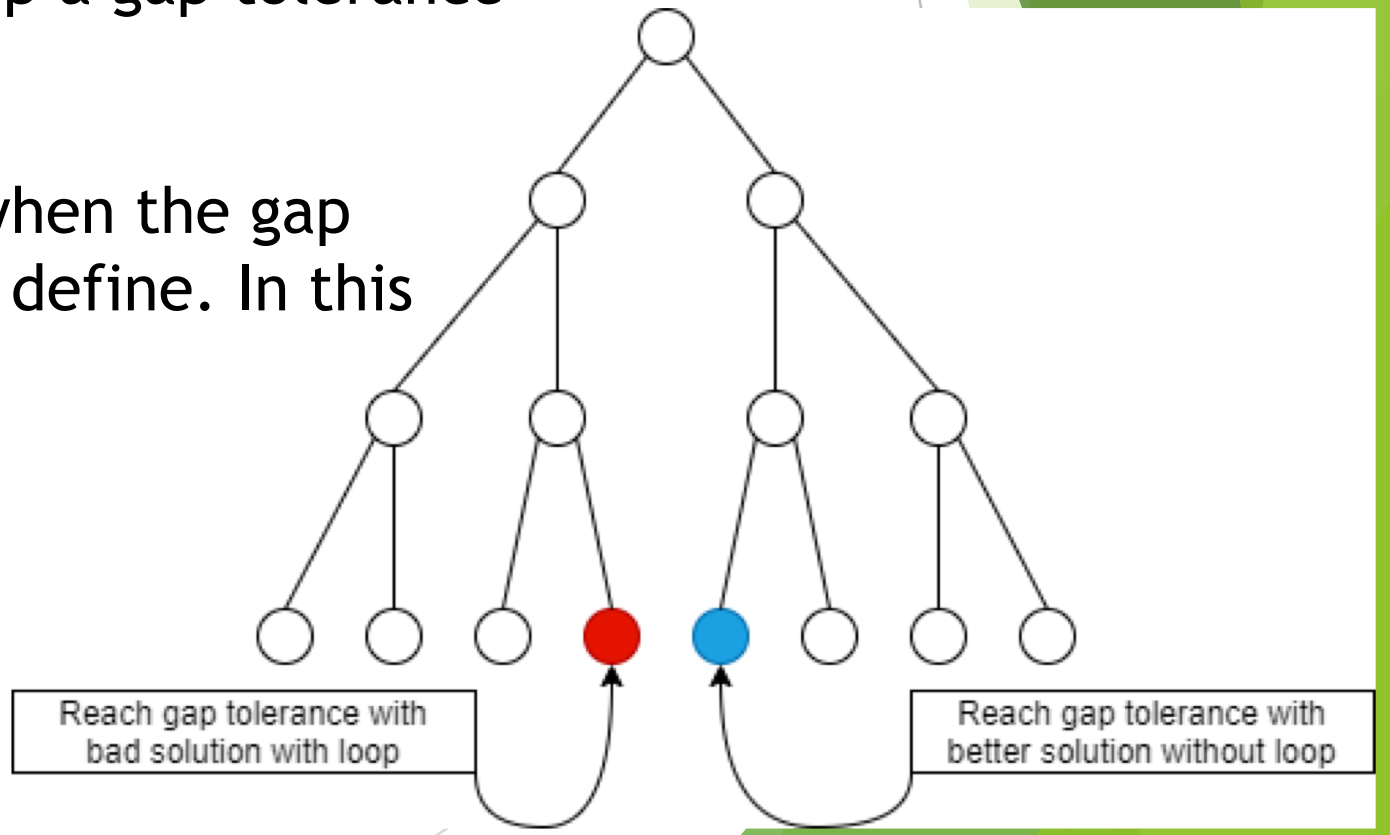
|best bound - best Integer| / (10 ^ -10 + |best Integer|)

- Best bound : The best solution in region constrained for any region that has not yet been eliminated from the search space.
  - Splitting the search space into sub-regions and search an optimal solution.
  - Remove sub-regions that can't possibly contain the optimal integer-feasible solution.
  - The possible solution to this modified problem is the best bound for the region.

Looping problem (Gap):

There are many solutions while solving a quadratic optimization problem. Since the size of the solution is too large, **waiting the optimal value to converge takes time**. Thus, we need to set up a gap tolerance to force the program to stop.

CPLEX will **randomly choose one** when the gap tolerance level reach the value we define. In this case, loop problem may happen.

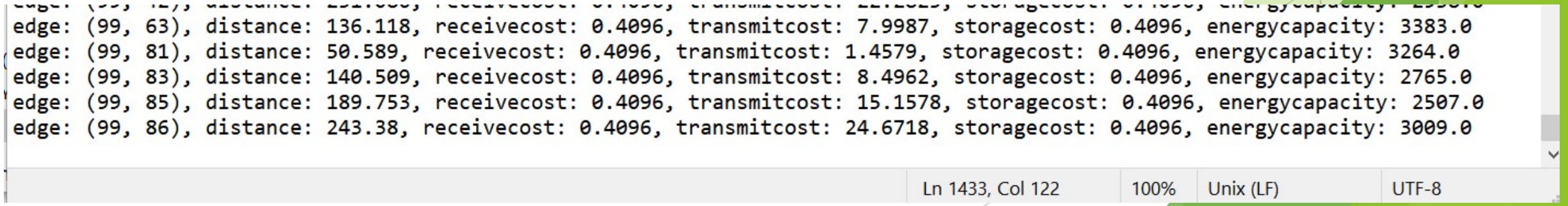


Time consuming issue:

The nodes' amount in the algorithm's tree depends on **how many edges** we have in our graph.

If we have 1432 edges (one edge = one variable = one layer of the tree), there will have **at most $2^{1432} - 1$ nodes in the tree** to go over. Although cutting can eliminate some branches, there are still many nodes to calculate.

Due to the large model, calculating the most optimal solution (0% gap tolerance) takes time and may encounter memory problem.



We run almost 47243 sec (13 hr) and still cannot reach the most optimal solution.

Error 3019 means that one subproblem encountered an **unrecoverable numerical issue**. This failure can be due to a limit (for example, an **iteration limit**) or due to numeric trouble (such as the **calculated value cannot converge**).