Nutmeg: a MIP and CP Hybrid Solver Using Branch-and-Check

Edward Lam[1], Graeme Gange[1], Peter J. Stuckey[1], Pascal Van Hentenryck[2], Jip J. Dekker[1]

[1]Monash University

[2]Georgia Institute of Technology

Decomposition at 70: In Honor of Jacques Desrosieres and His Legacy
11 February 2021

Overview

- A generic method to find combinatorial Benders cuts.
- Based logic-based Benders decomposition.
    - Generalizes classical Benders decomposition to discrete subproblems.
    - No specific form of cuts due to its generality. User must derive cuts valid for their problem.
    - Otherwise, naive combinatorial Benders cuts: $x_1,\dots,x_n$ binary, $x_1 + x_2 + \dots + x_n \leq n - 1$
- Uses propagation (aka inference) and conflict analysis (aka conflict-driven clause learning) from constraint programming and Boolean satisfiability (SAT).
- Central idea: find a set of values to some master-problem variables that implicate infeasibility in the subproblem.

Example

Vehicle Routing Problem with Time Windows

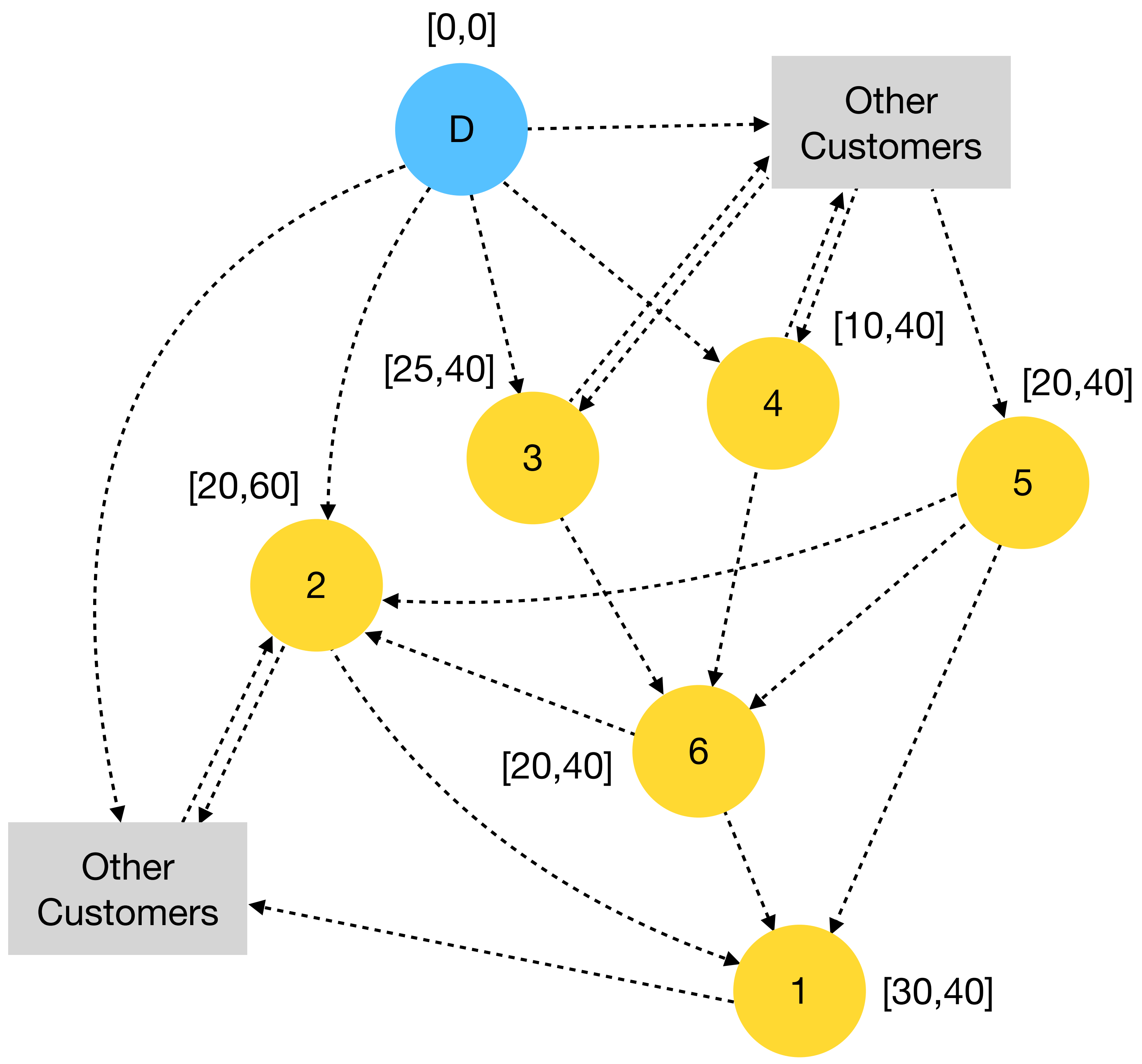
- Familiar problem to illustrate the method.
- Obviously not the best way to solve the VRPTW.
- Master problem has arc variables and network flow constraints.
- Travel time and time window constraints moved into the Benders subproblem.
- Ignore load in this example.

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)

Network



Branch-and-Bound Tree

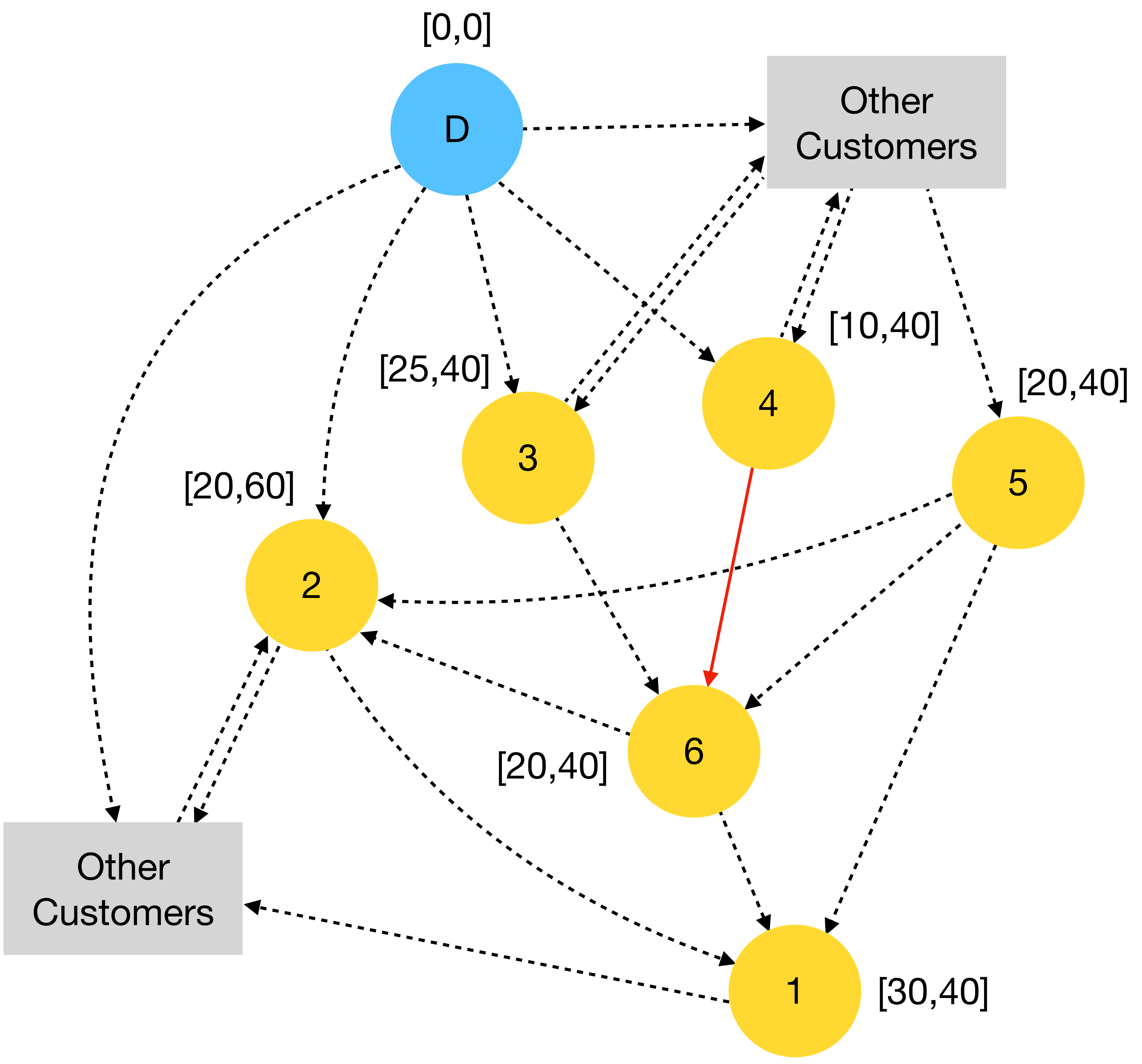
Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data	Decision (branching)	Propagation (inferred)

$x_{4,6} = 0$

Network



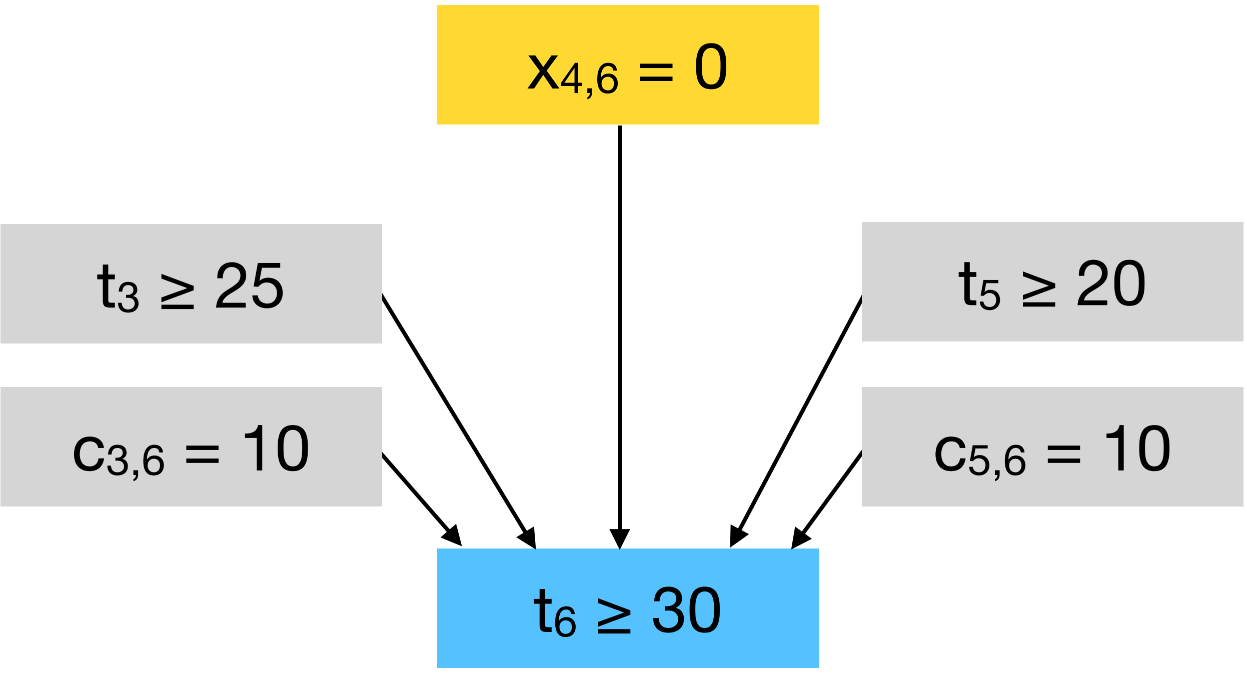
Branch-and-Bound Tree

$x_{4,6} = 0$

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$t_3 \geq 25$

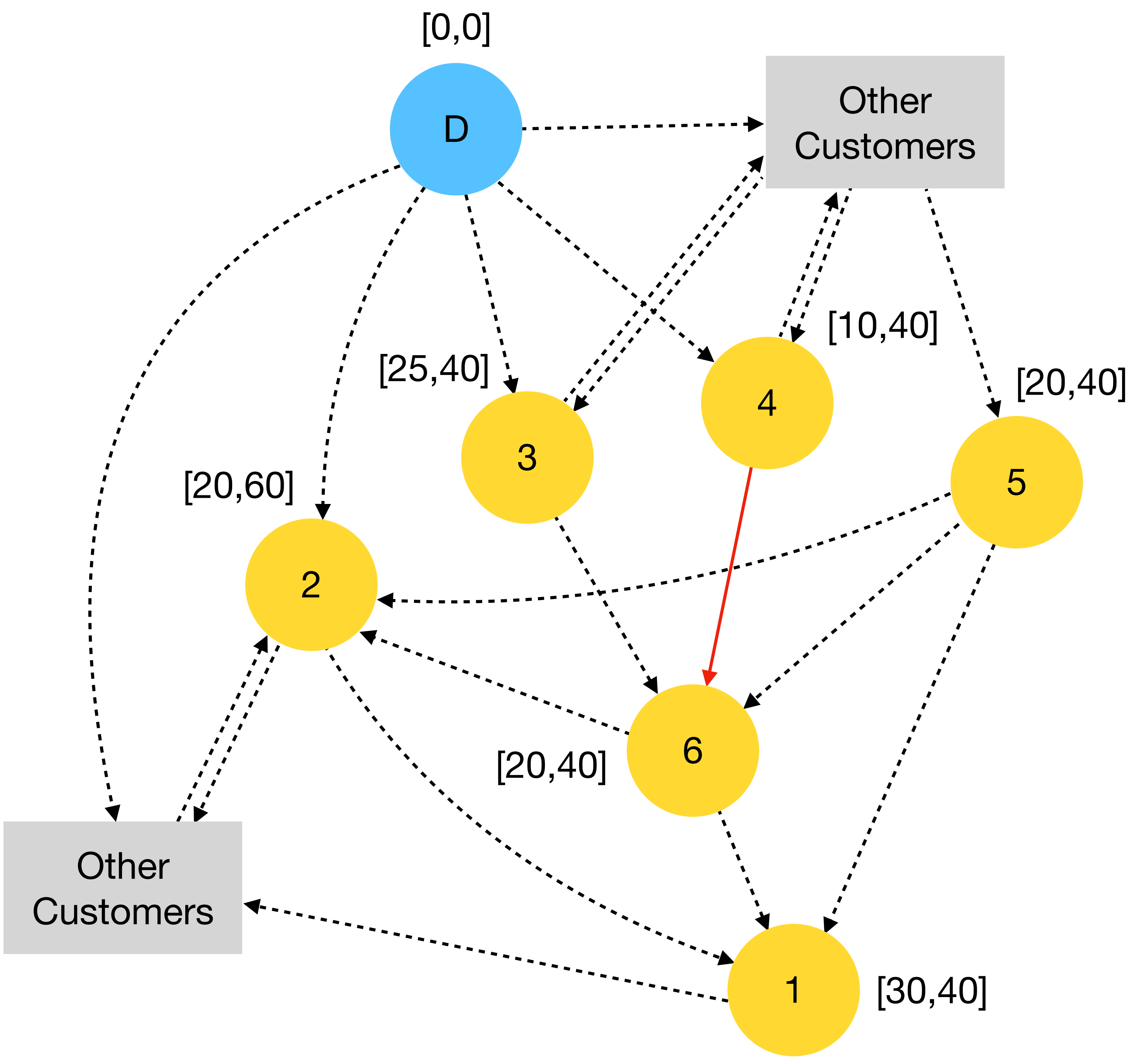
$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$t_6 \geq 30$

Network



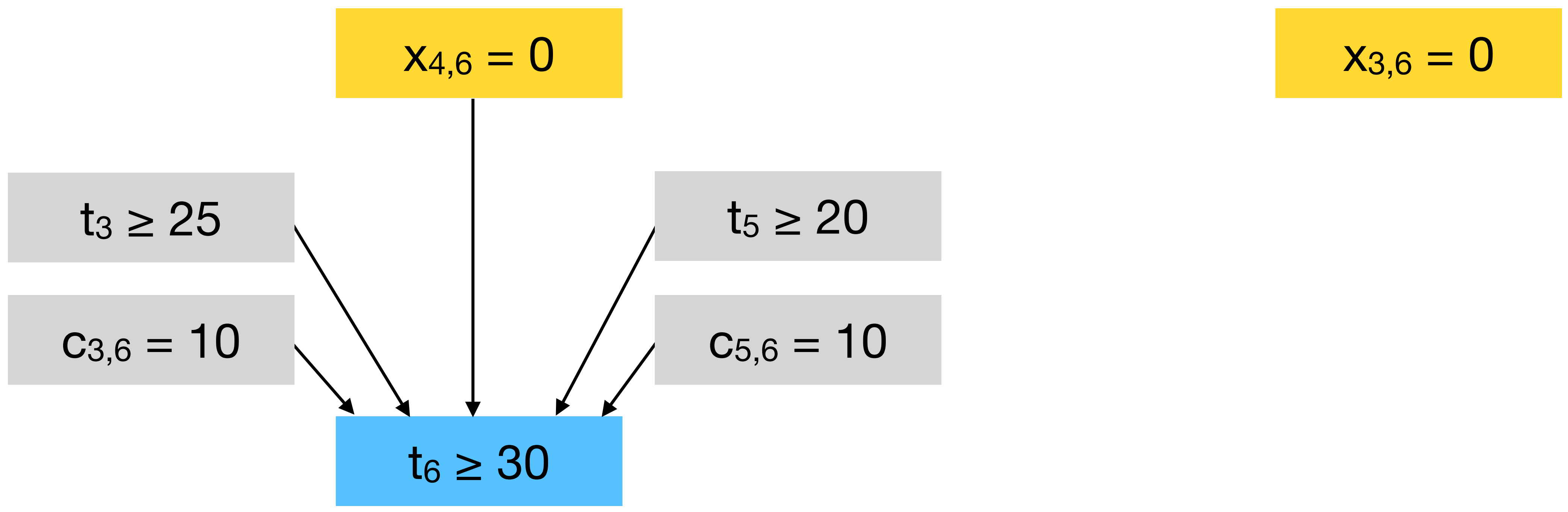
Branch-and-Bound Tree

$x_{4,6} = 0$

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

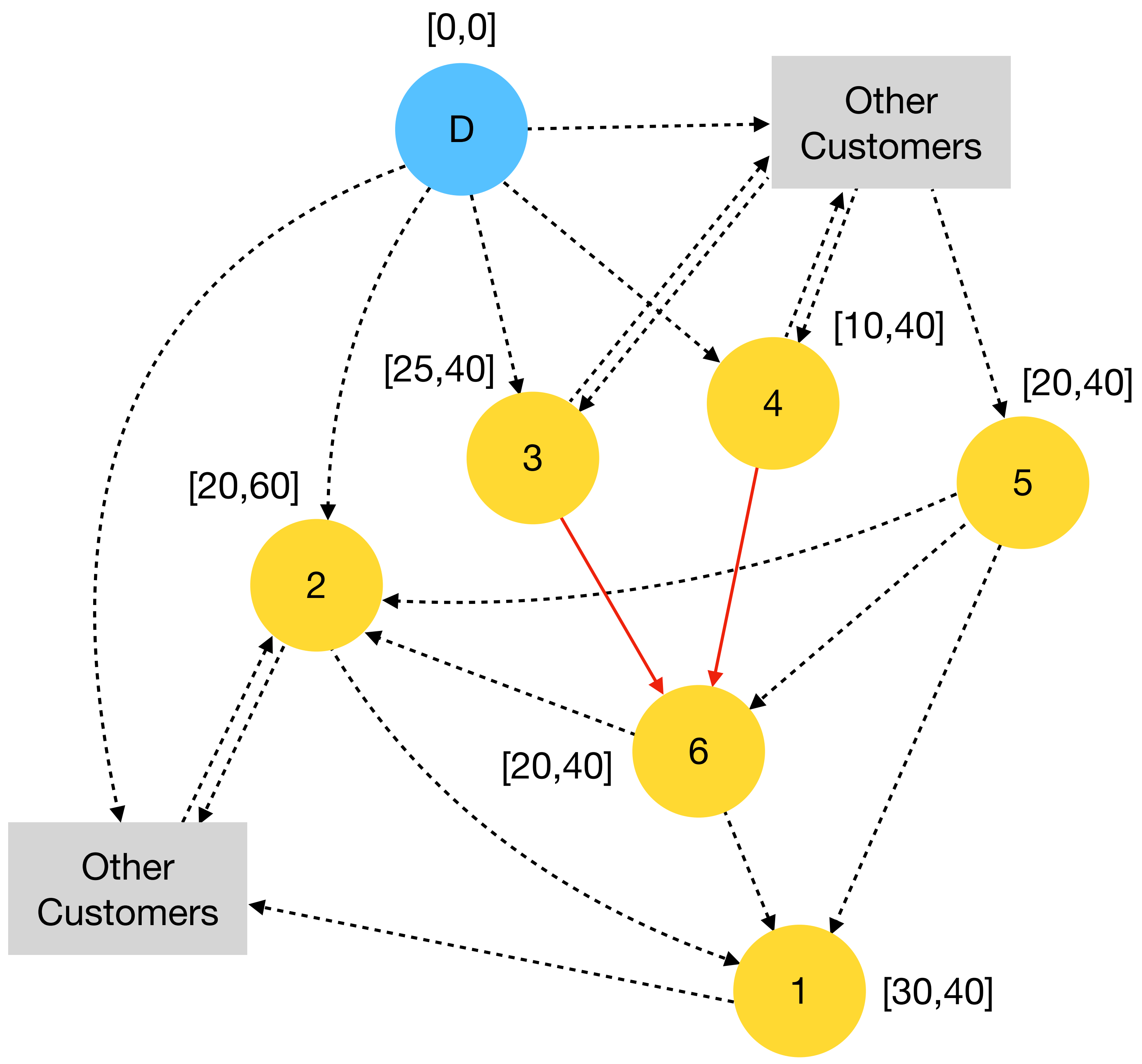
$c_{3,6} = 10$

$t_5 \geq 20$

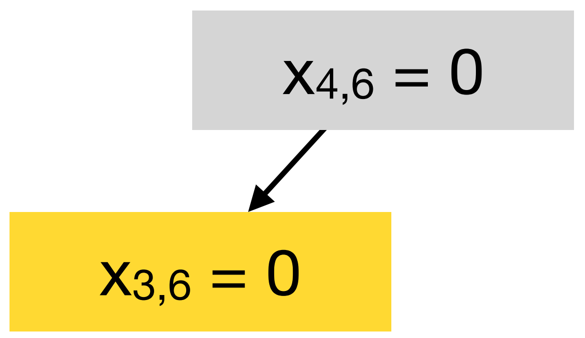
$c_{5,6} = 10$

$t_6 \geq 30$

Network



Branch-and-Bound Tree



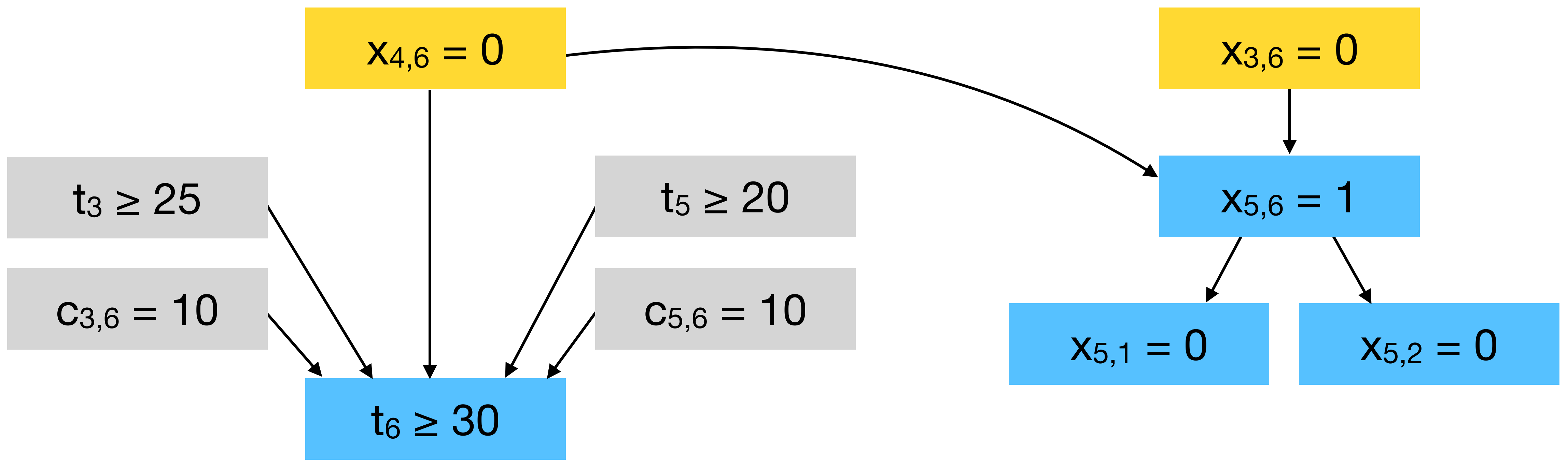
$x_{4,6} = 0$

$x_{3,6} = 0$

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$t_5 \geq 20$

$x_{5,6} = 1$

$c_{3,6} = 10$

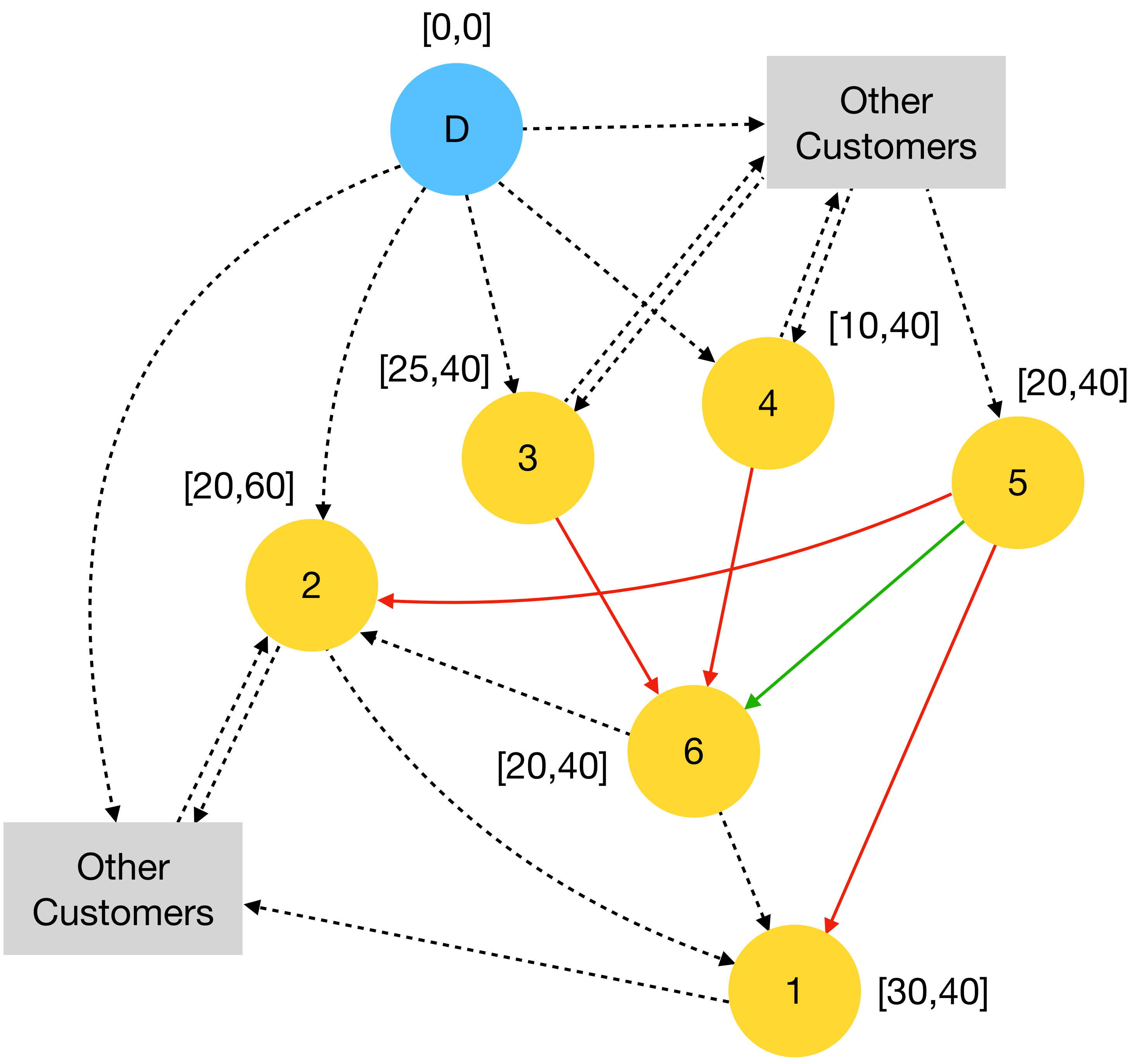
$c_{5,6} = 10$

$x_{5,1} = 0$

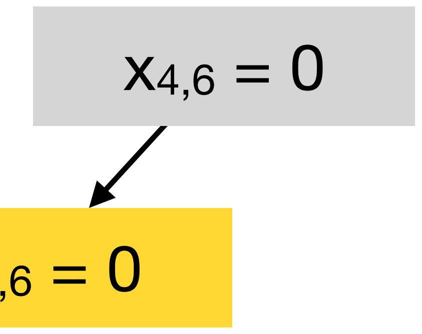
$x_{5,2} = 0$

$t_6 \geq 30$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

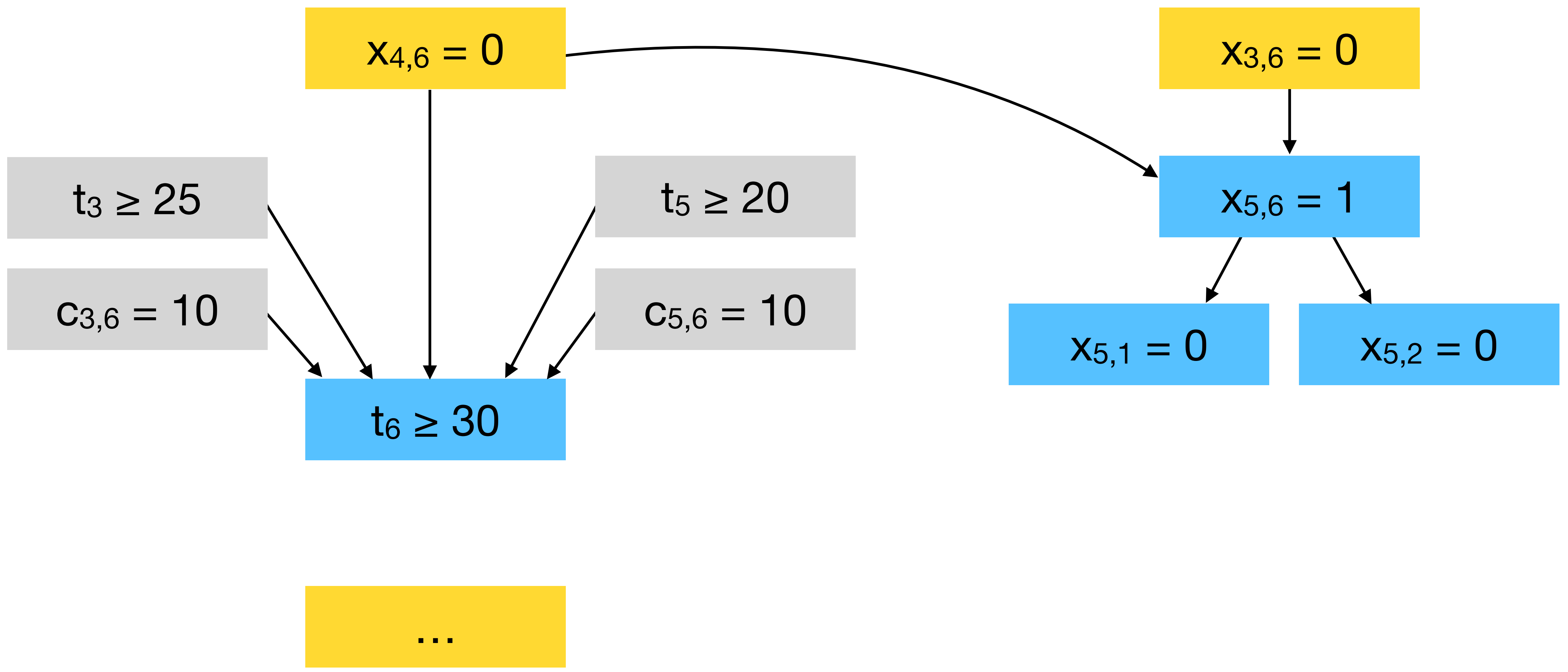
Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data

Decision (branching)

Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$x_{5,6} = 1$

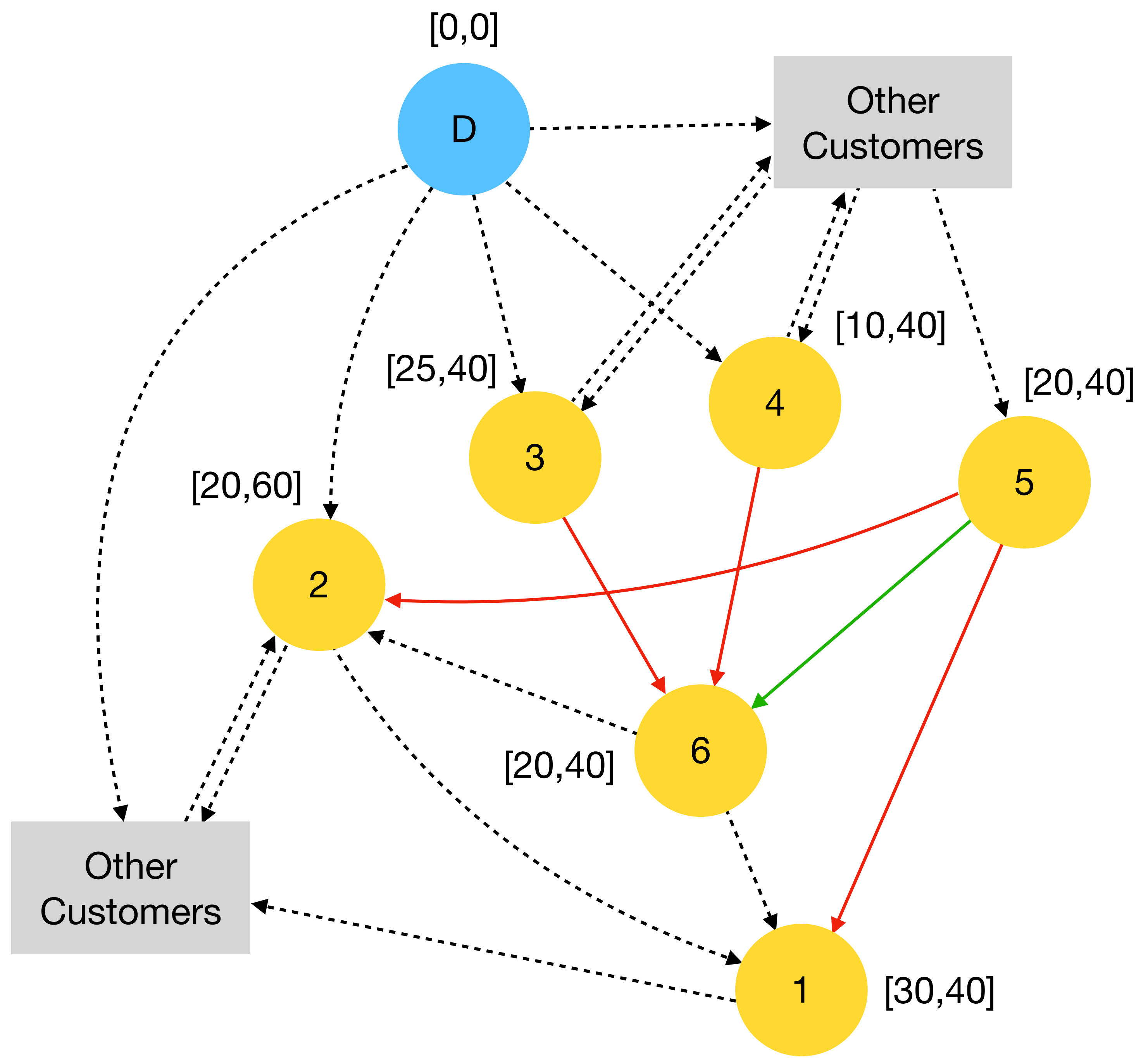
$x_{5,1} = 0$

$x_{5,2} = 0$

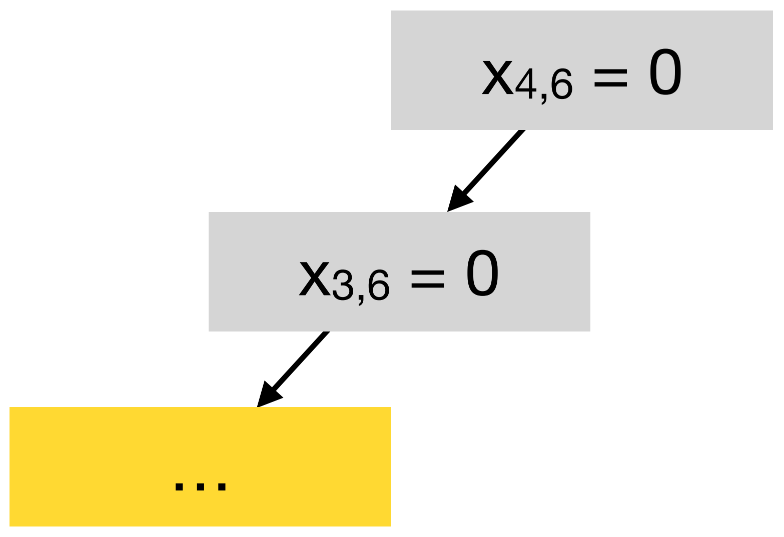
$t_6 \geq 30$

…

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

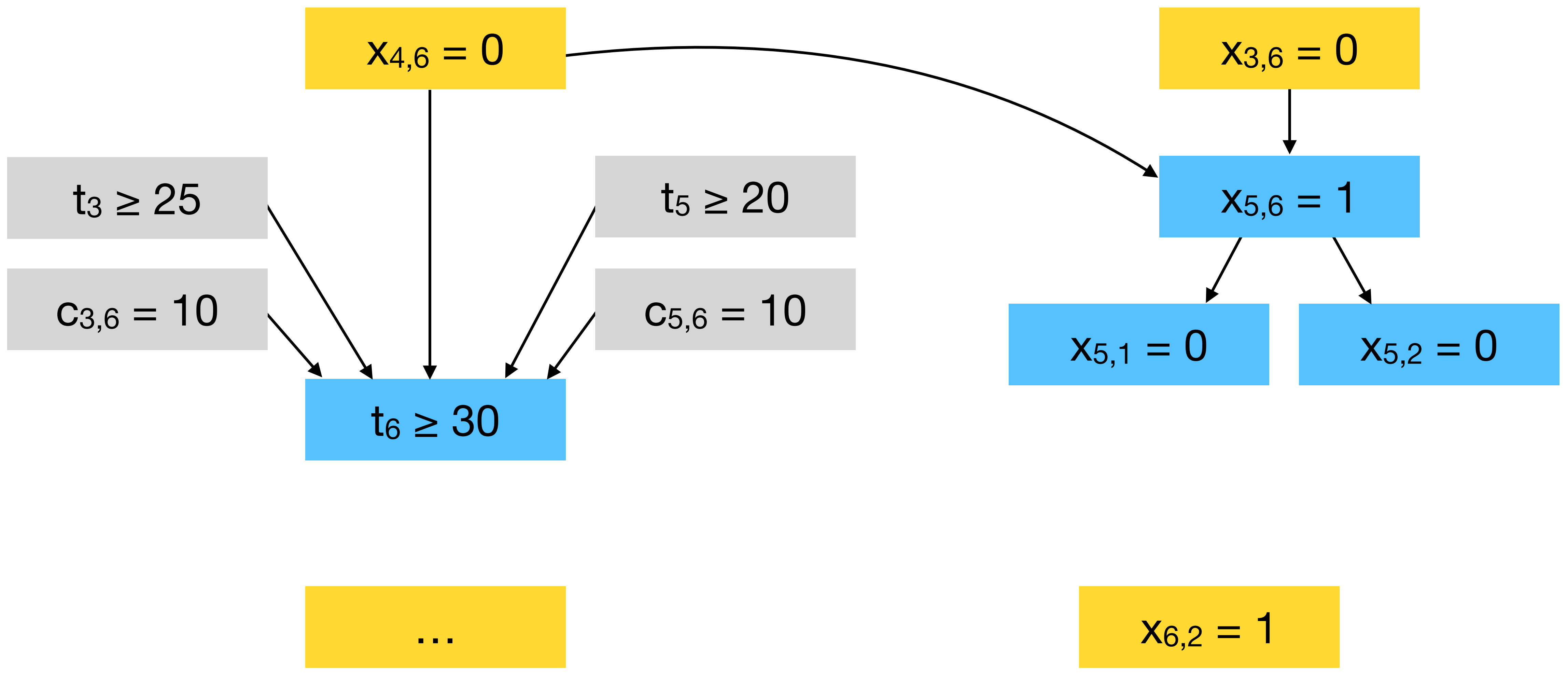
$x_{3,6} = 0$

…

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$t_6 \geq 30$

$x_{5,6} = 1$

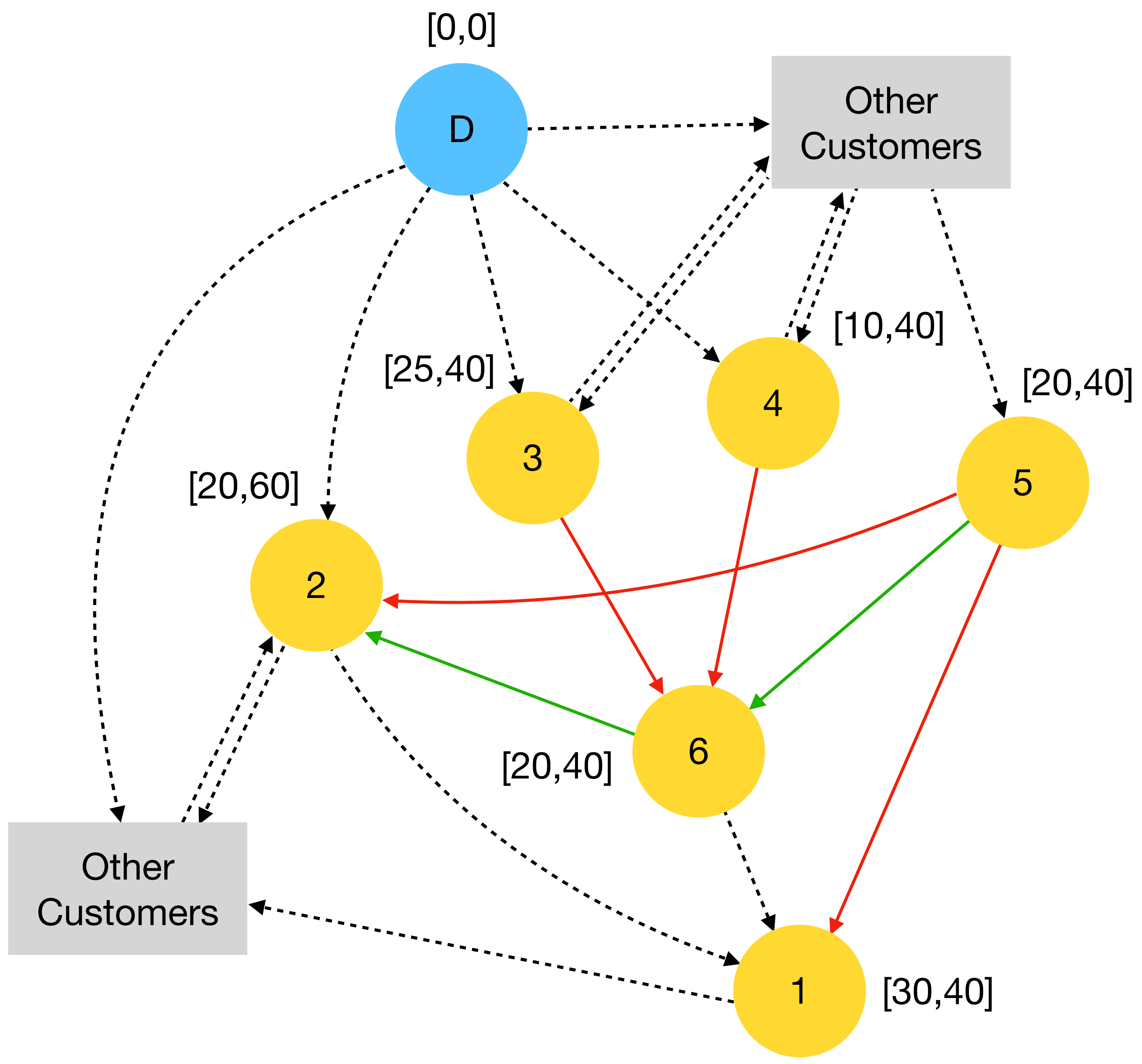
$x_{5,1} = 0$

$x_{5,2} = 0$

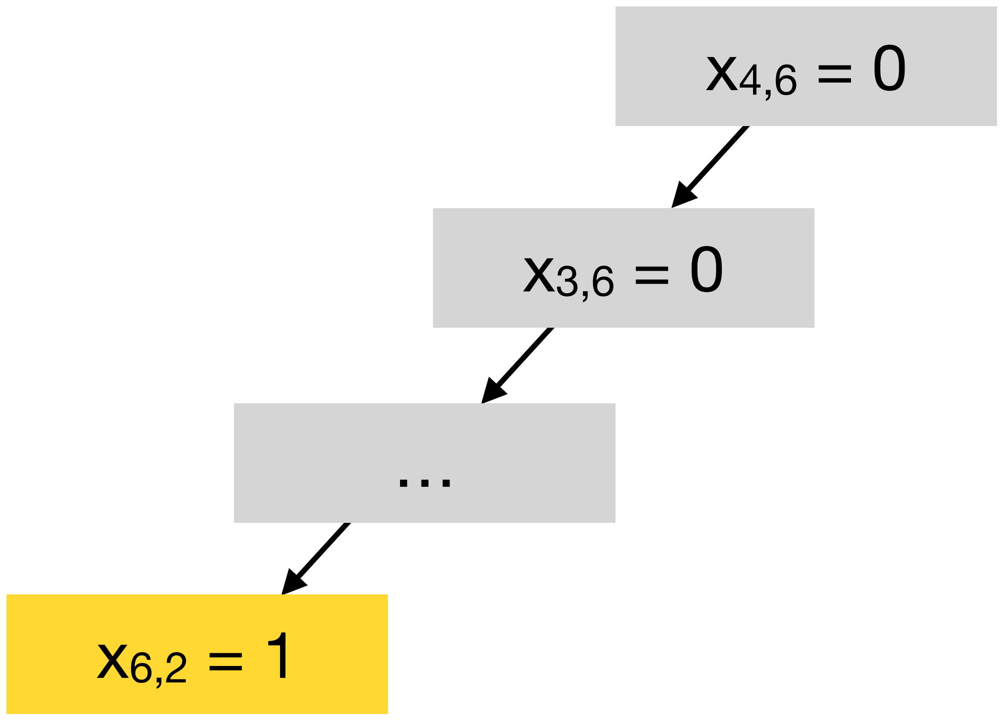
…

$x_{6,2} = 1$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

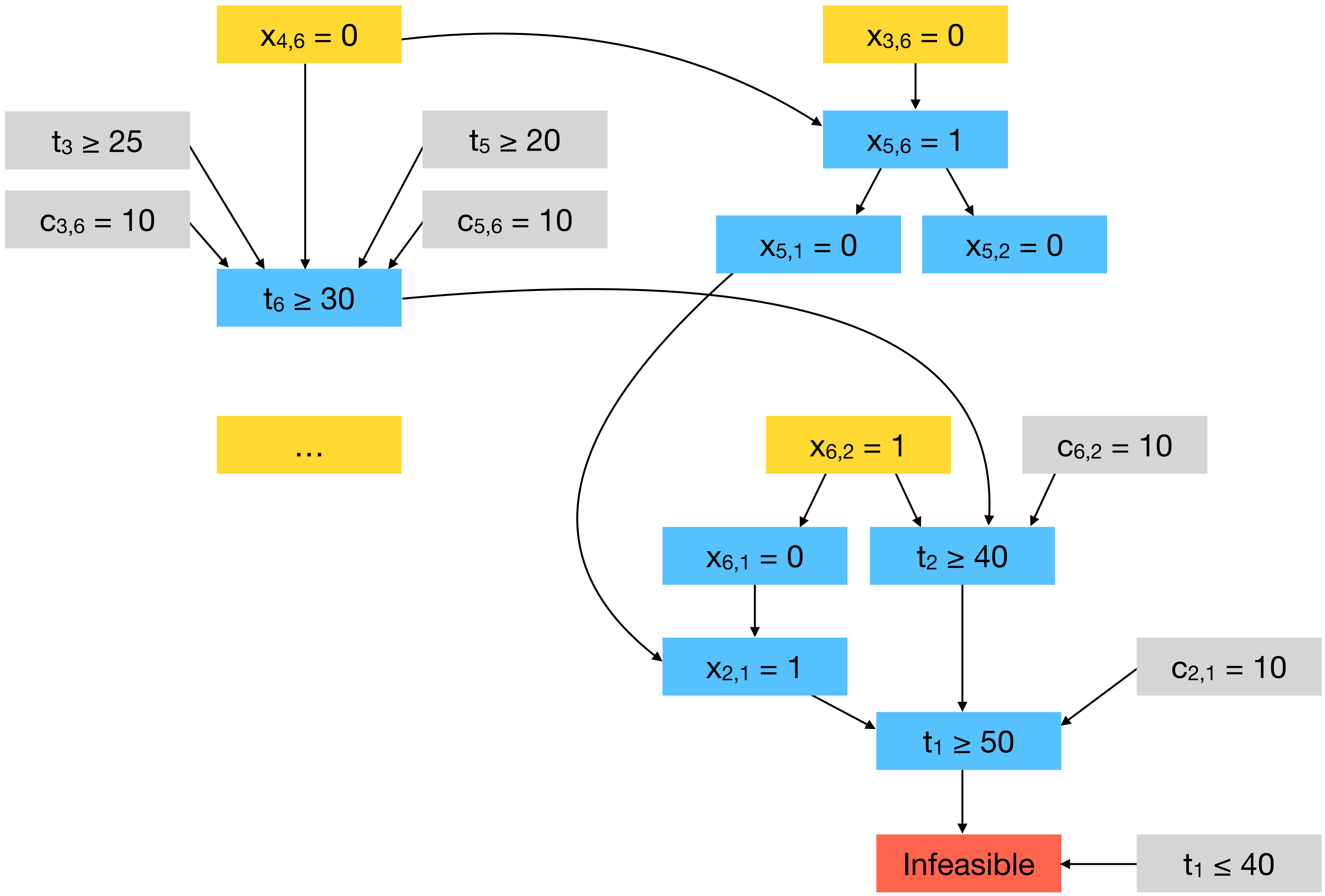
…

$x_{6,2} = 1$

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$x_{5,6} = 1$

$x_{5,1} = 0$

$x_{5,2} = 0$

$t_6 \geq 30$

…

$x_{6,2} = 1$

$c_{6,2} = 10$

$x_{6,1} = 0$

$t_2 \geq 40$

$x_{2,1} = 1$

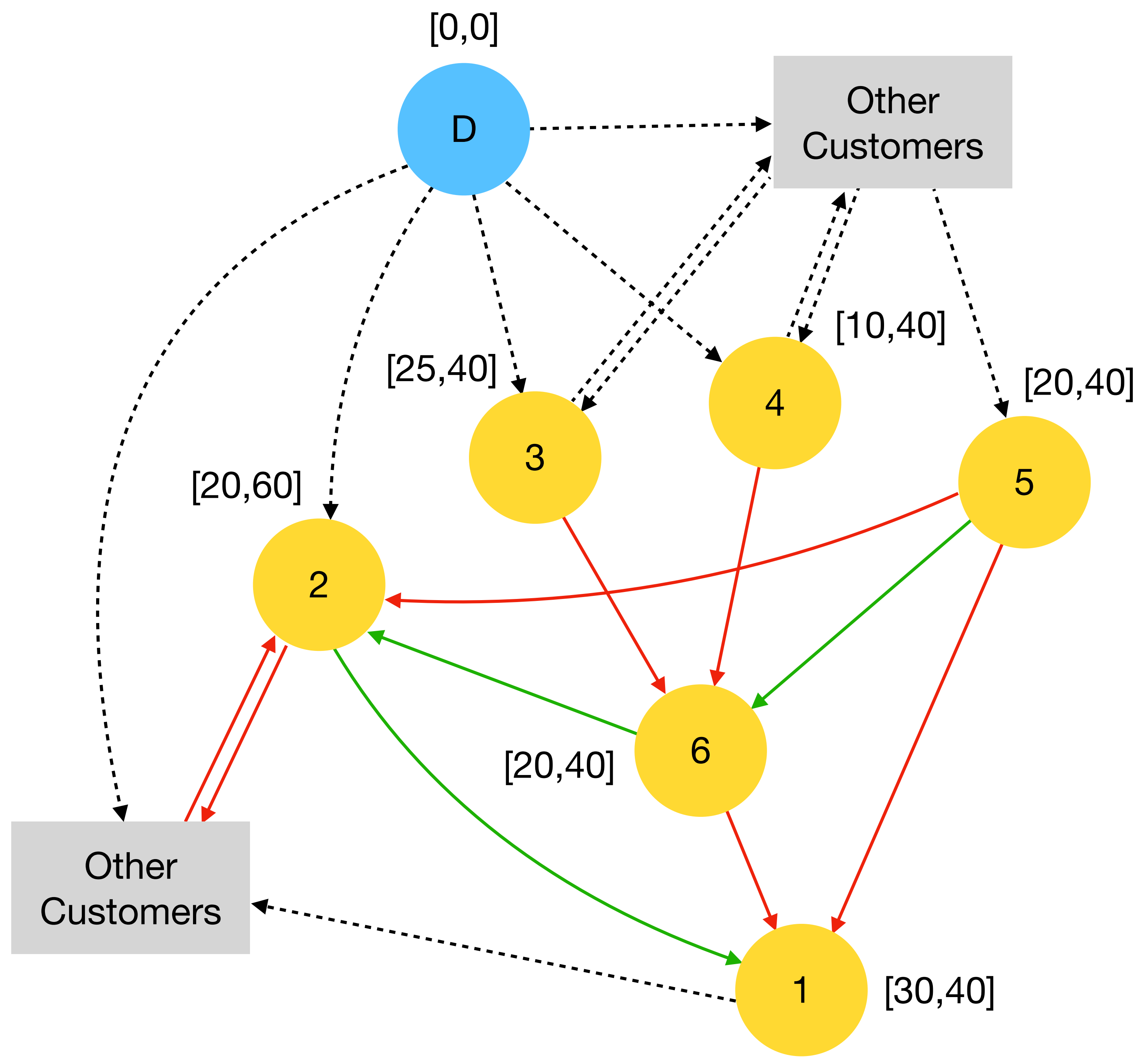
$c_{2,1} = 10$

$t_1 \geq 50$

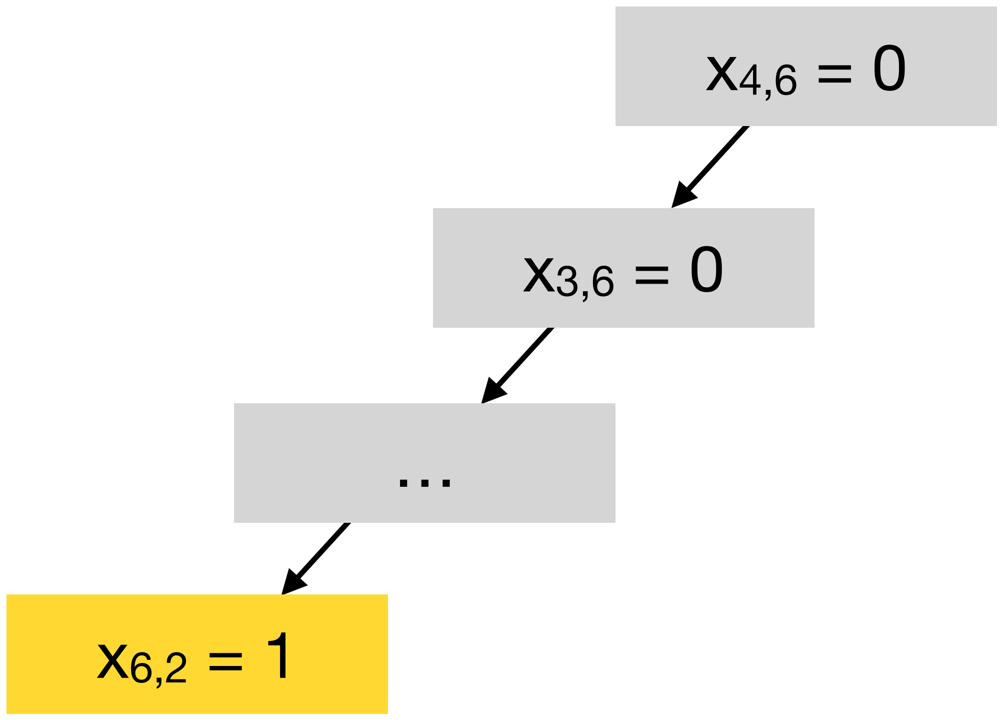
Infeasible

$t_1 \leq 40$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

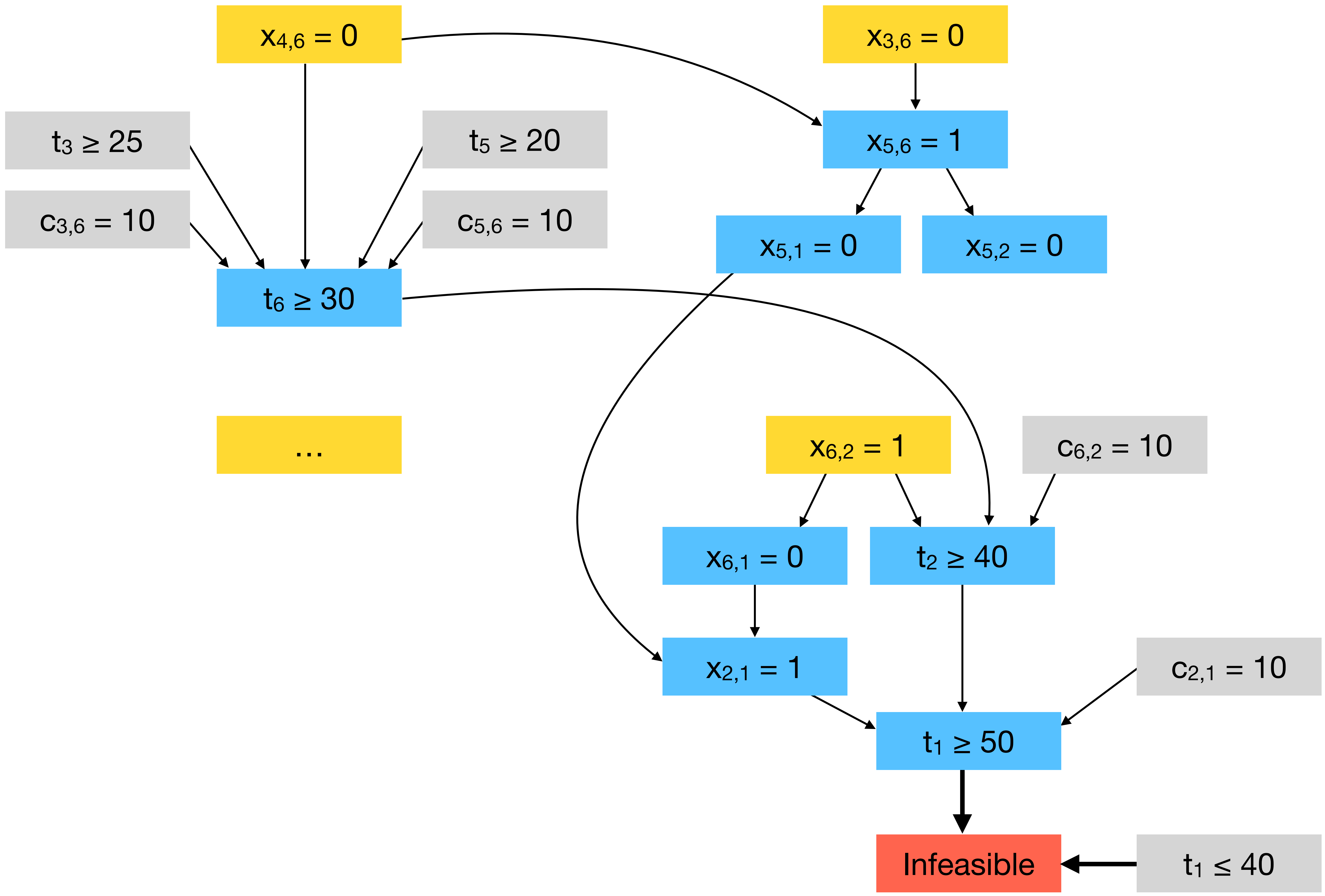
…

$x_{6,2} = 1$

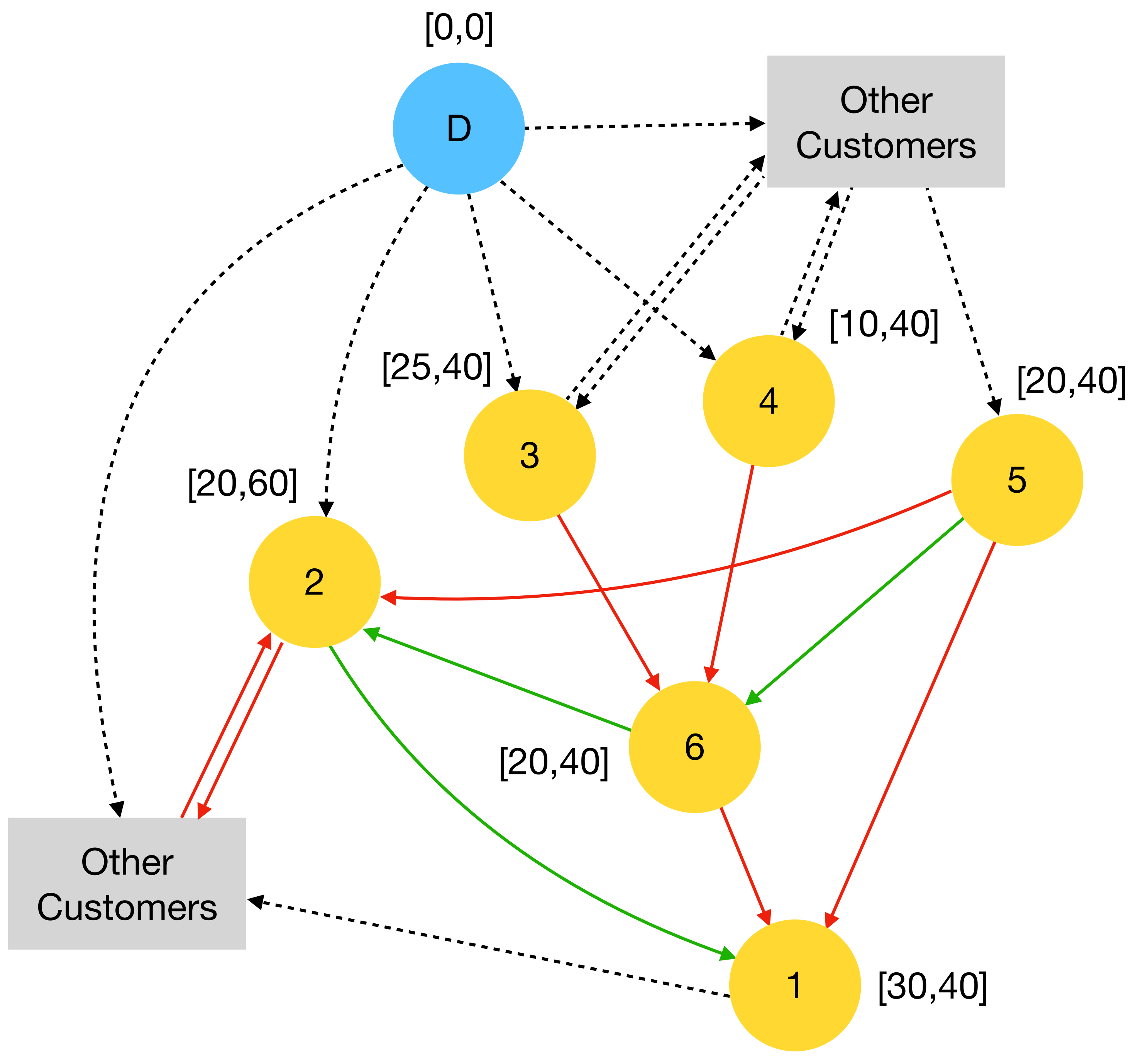
Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

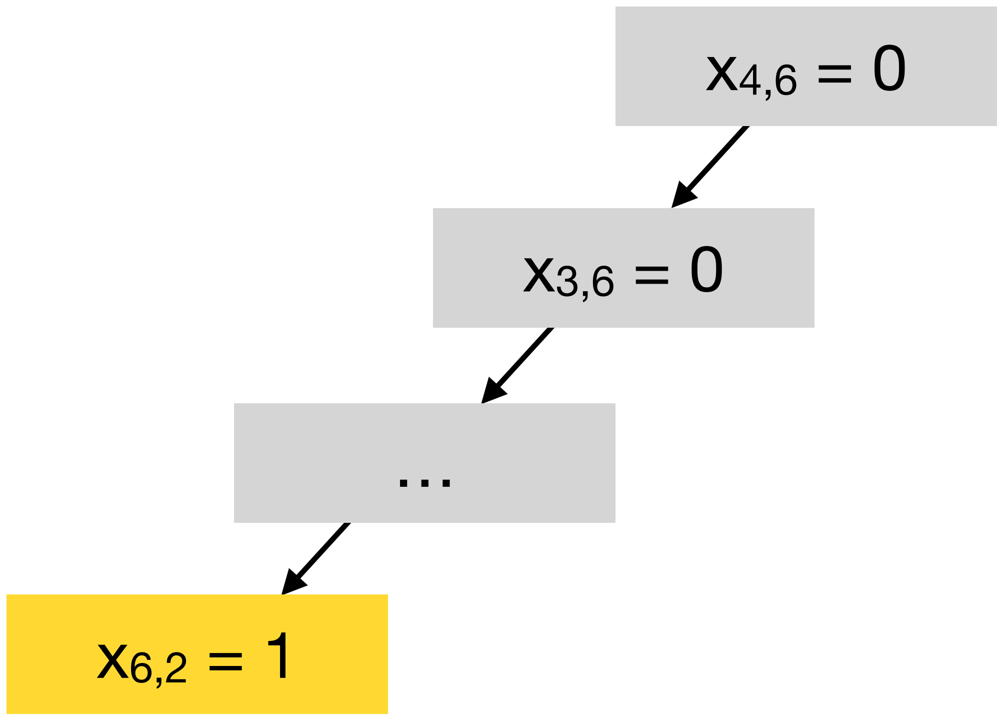
Data | Decision (branching) | Propagation (inferred)



Network



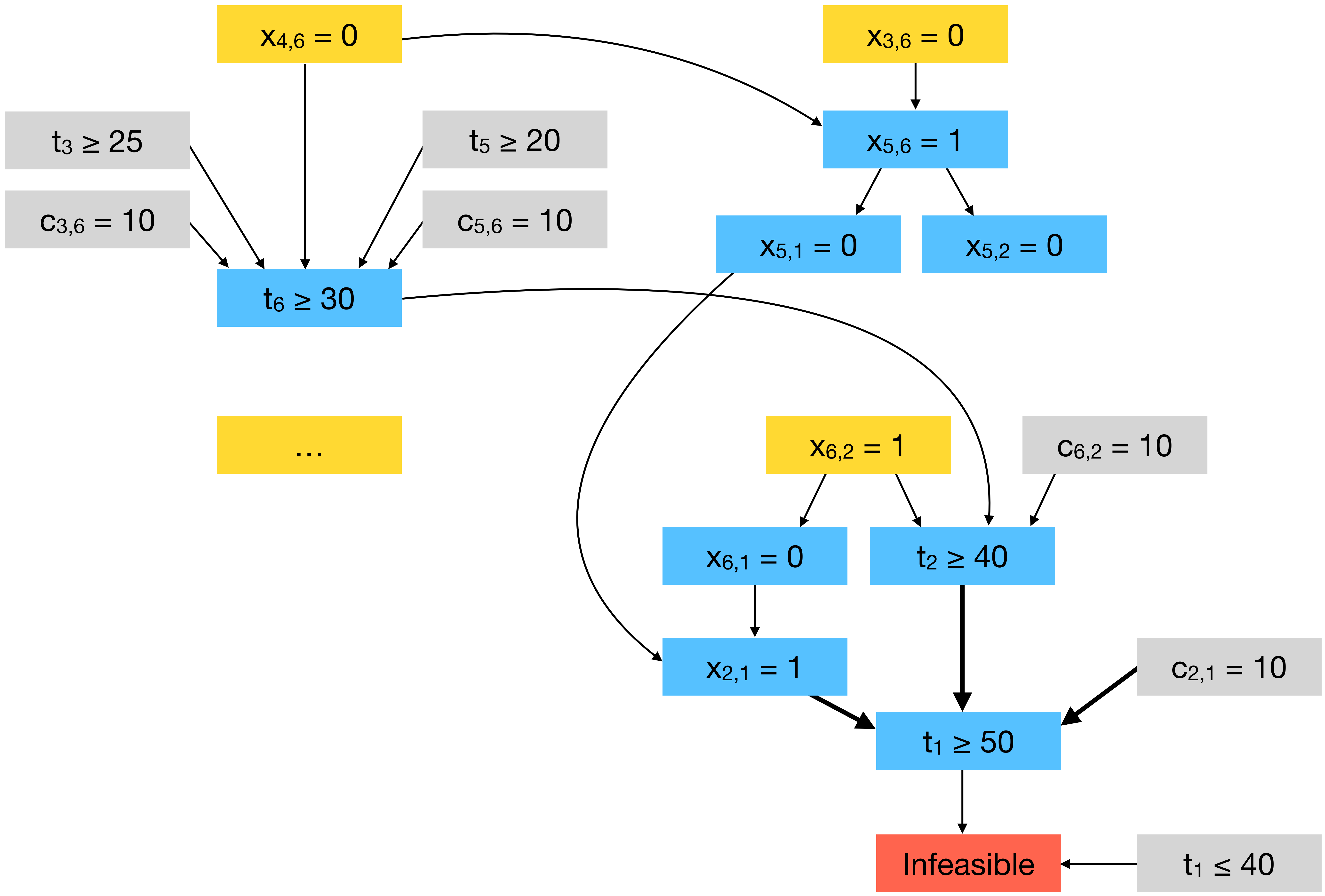
Branch-and-Bound Tree



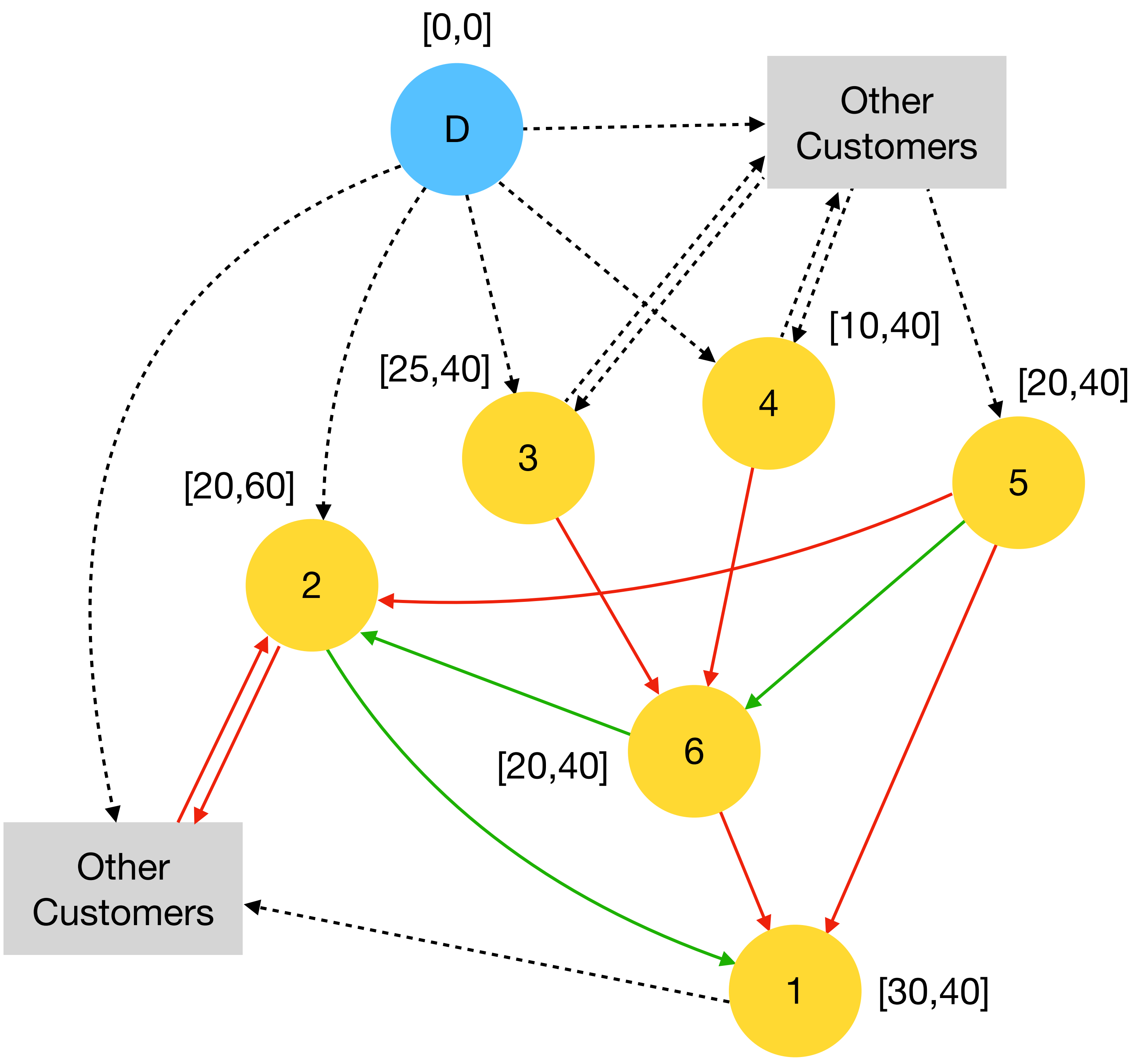
Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

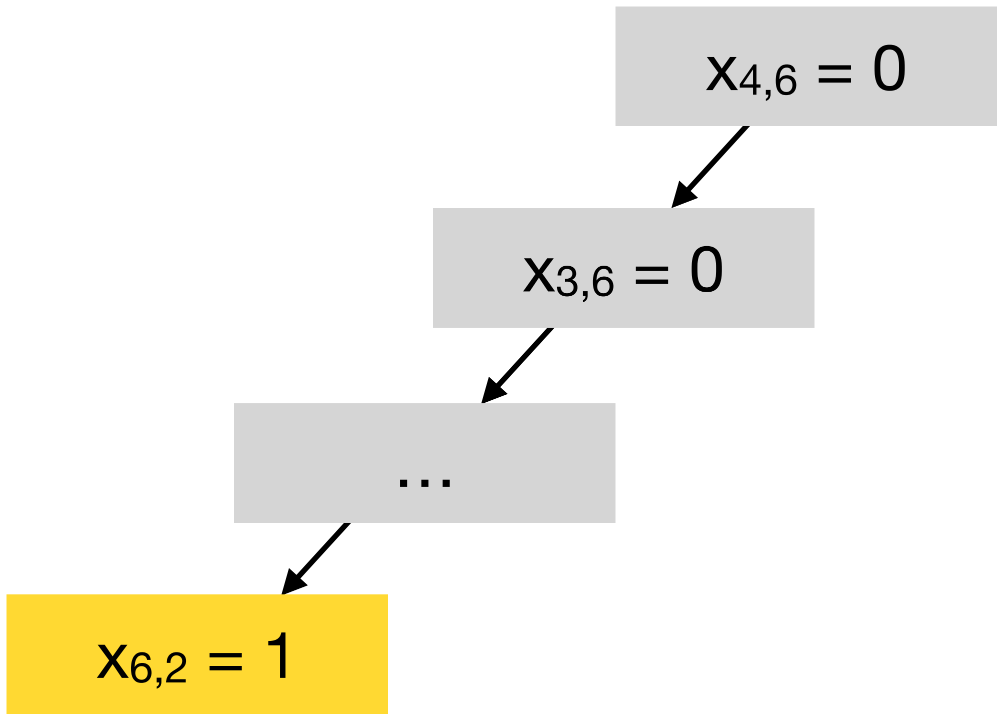
Data | Decision (branching) | Propagation (inferred)



Network



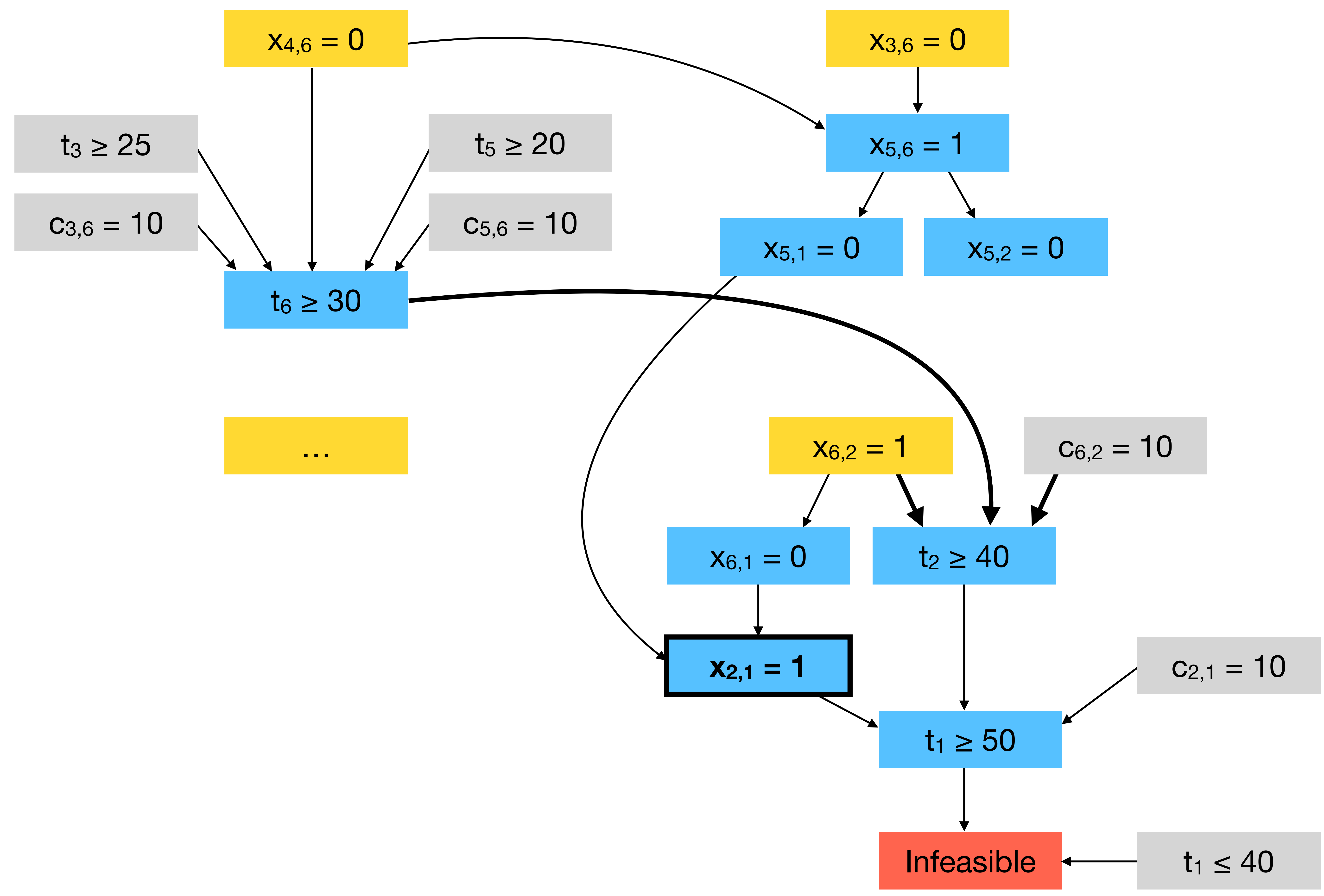
Branch-and-Bound Tree



Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$x_{5,6} = 1$

$x_{5,1} = 0$

$x_{5,2} = 0$

$t_6 \geq 30$

…

$x_{6,2} = 1$

$c_{6,2} = 10$

$x_{6,1} = 0$

$t_2 \geq 40$

$\mathbf{x_{2,1} = 1}$

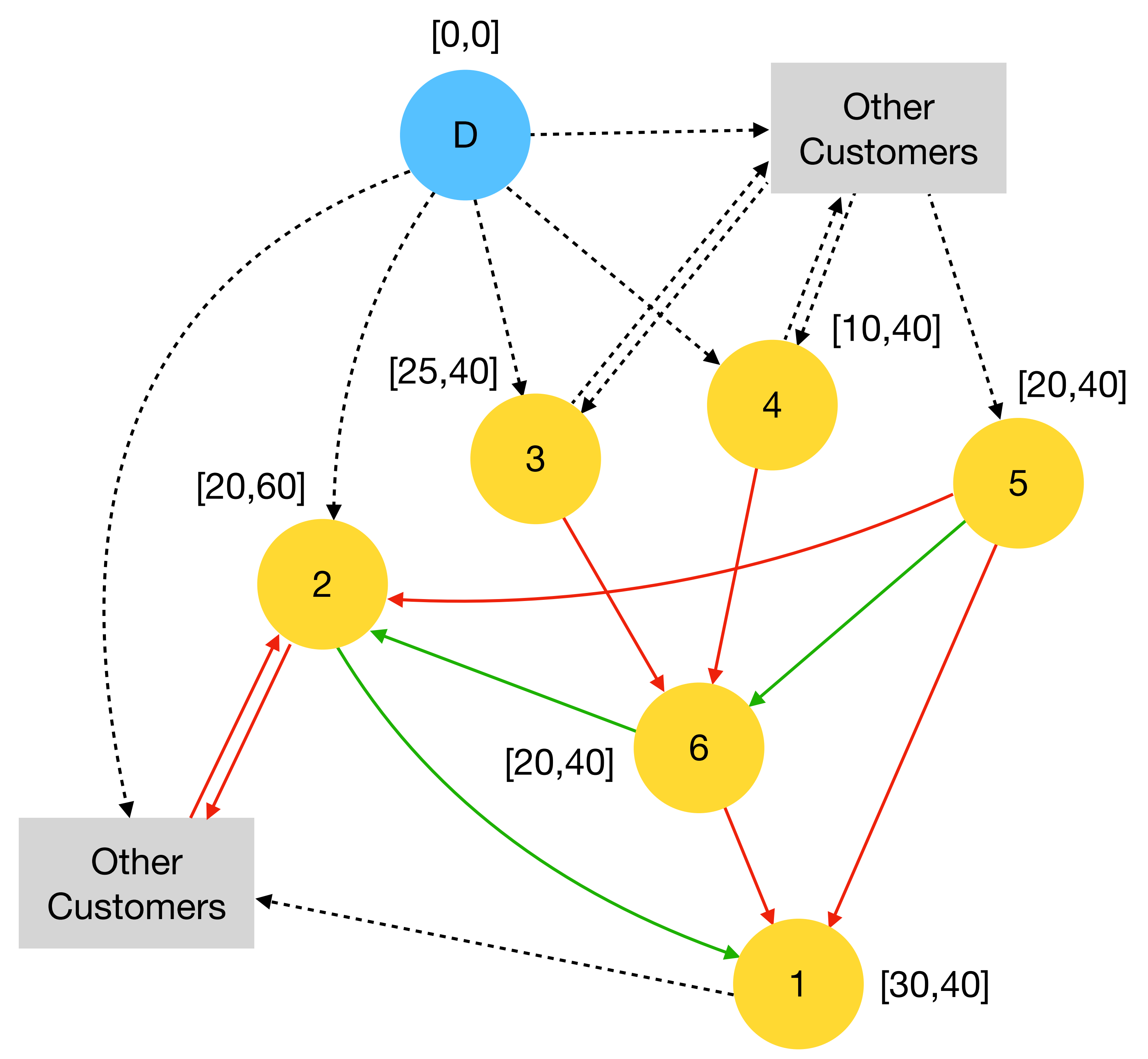
$c_{2,1} = 10$

$t_1 \geq 50$

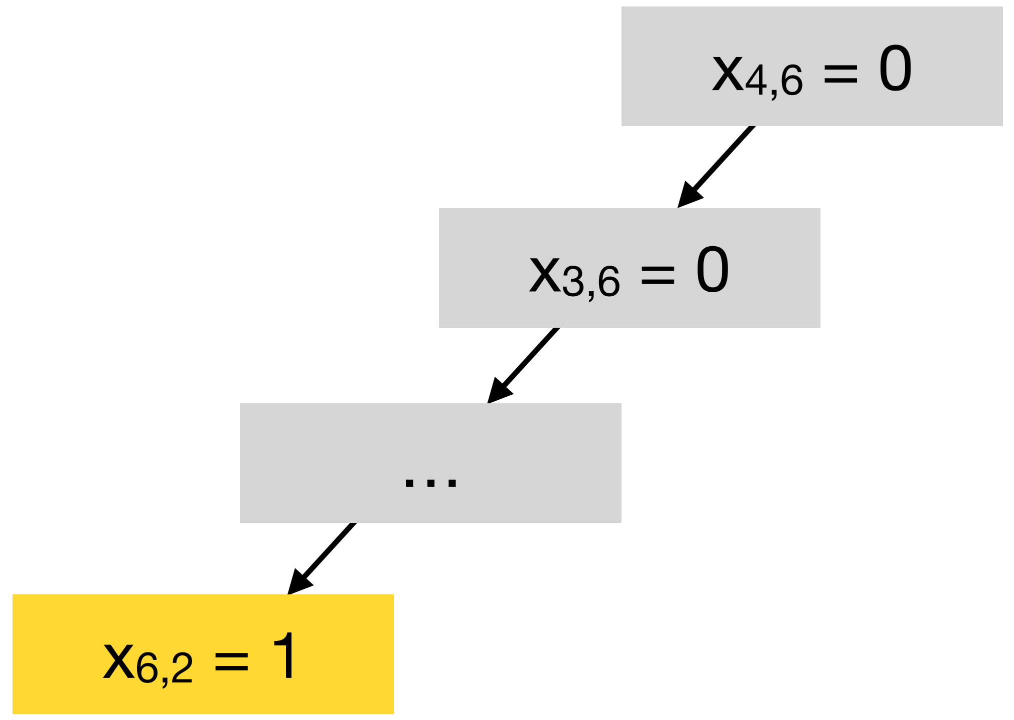
Infeasible

$t_1 \leq 40$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

…

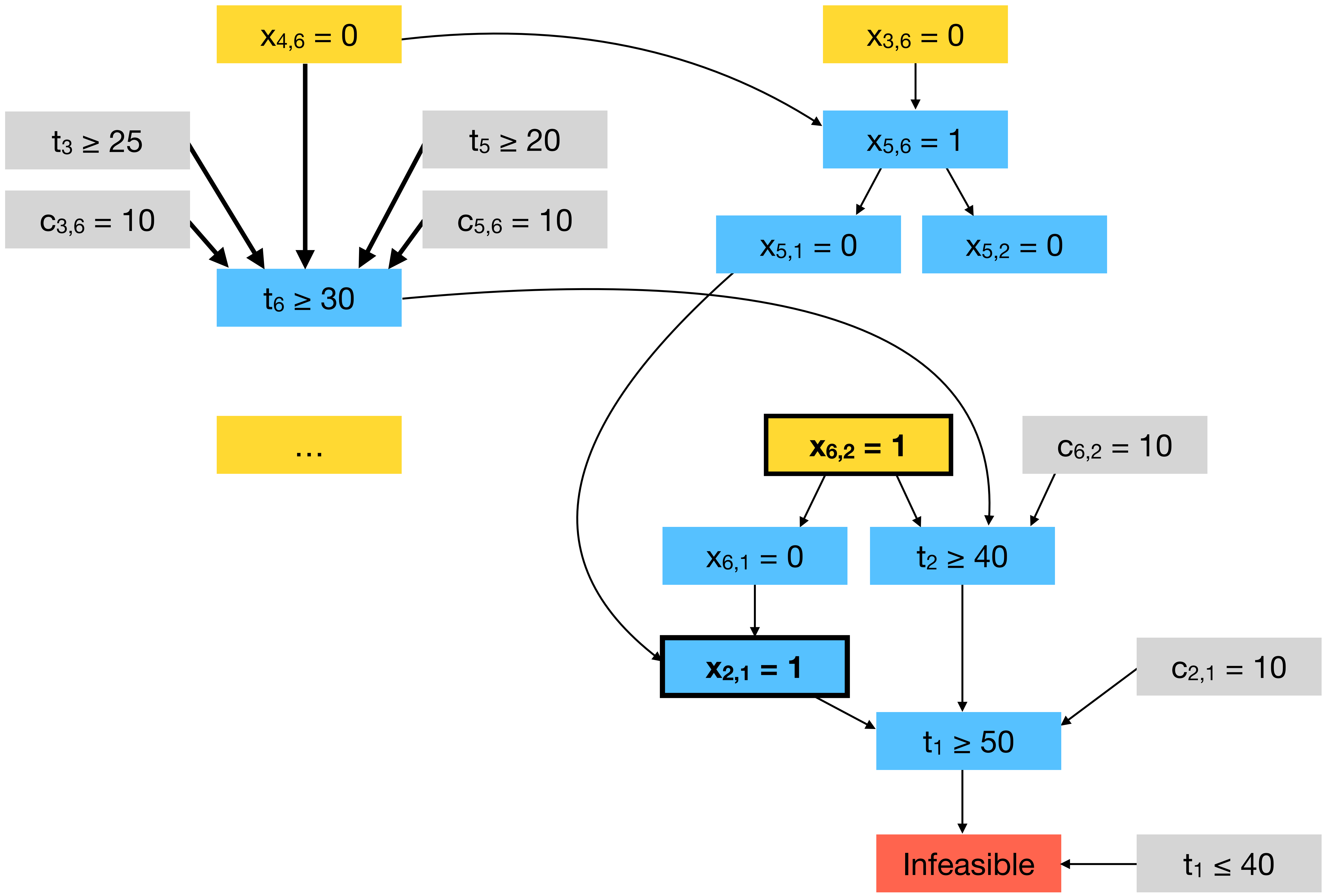
$x_{6,2} = 1$

Implication Graph

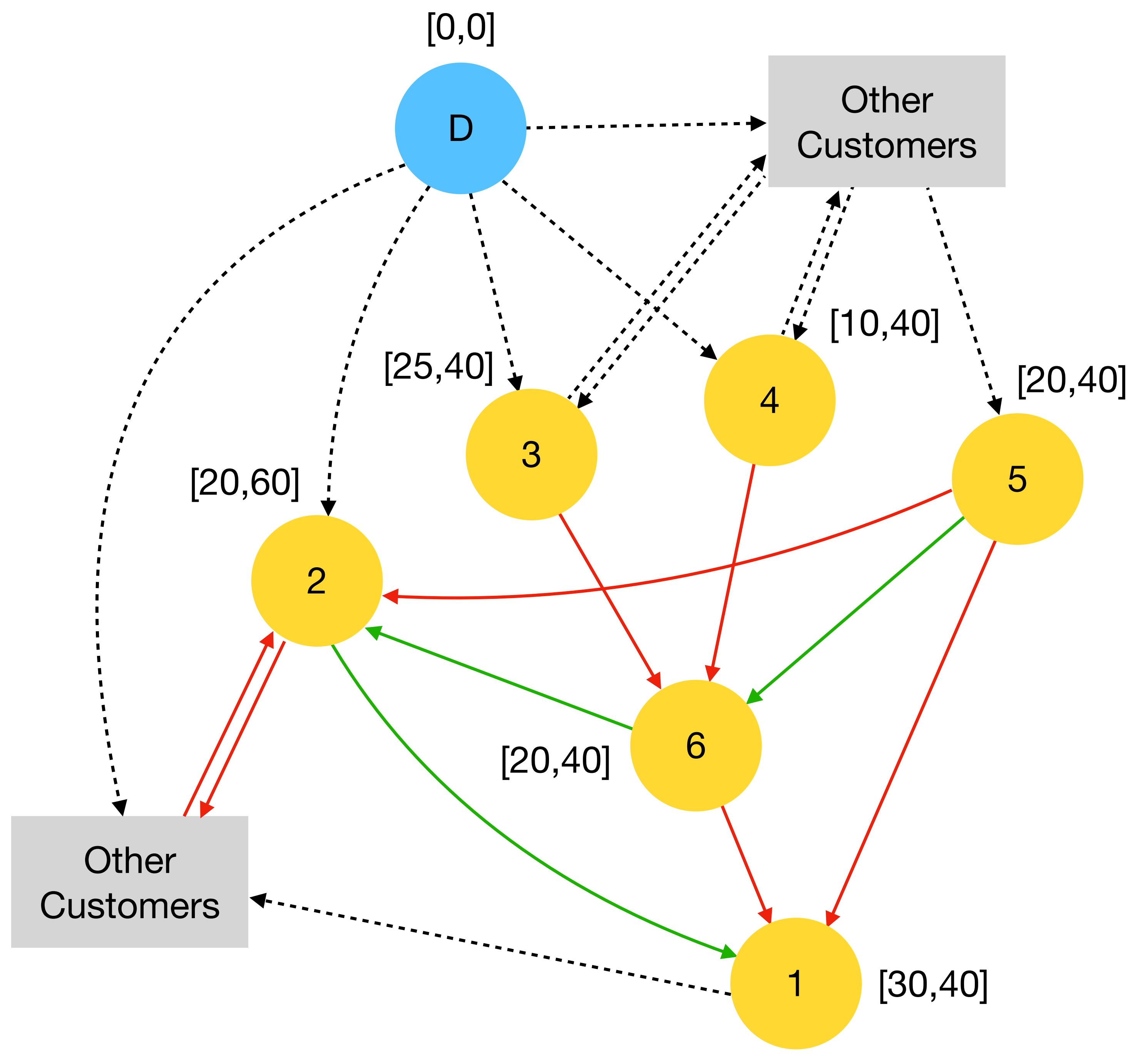
t_i: earliest time of starting service at vertex i

$c_{i,j}$: cost/travel time from i to j

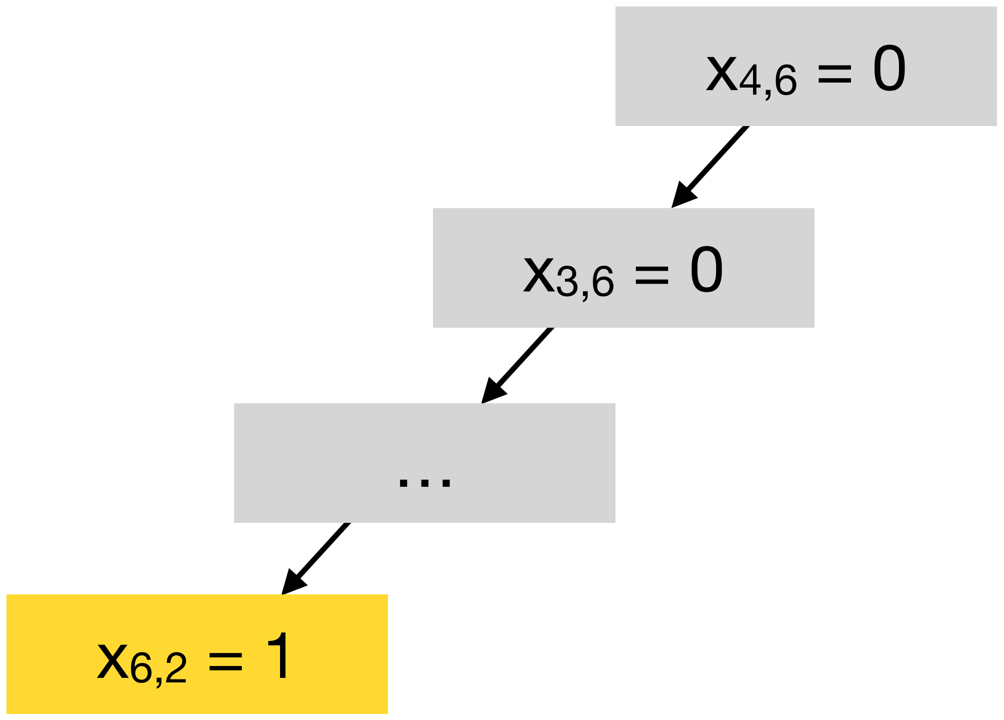
Data | Decision (branching) | Propagation (inferred)



Network



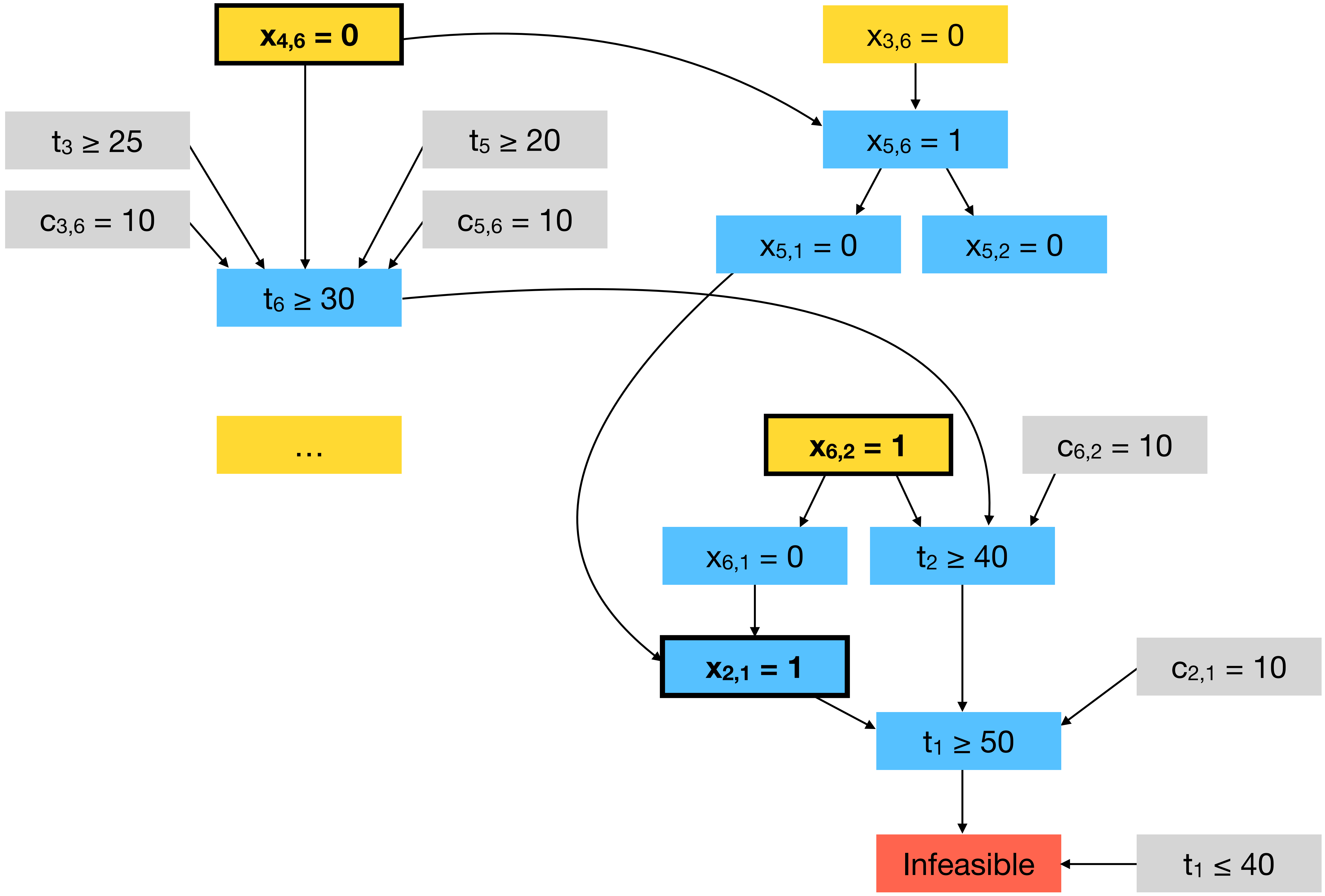
Branch-and-Bound Tree



Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$c_{3,6} = 10$

$t_5 \geq 20$

$c_{5,6} = 10$

$x_{5,6} = 1$

$x_{5,1} = 0$

$x_{5,2} = 0$

$t_6 \geq 30$

…

$x_{6,2} = 1$

$c_{6,2} = 10$

$x_{6,1} = 0$

$t_2 \geq 40$

$x_{2,1} = 1$

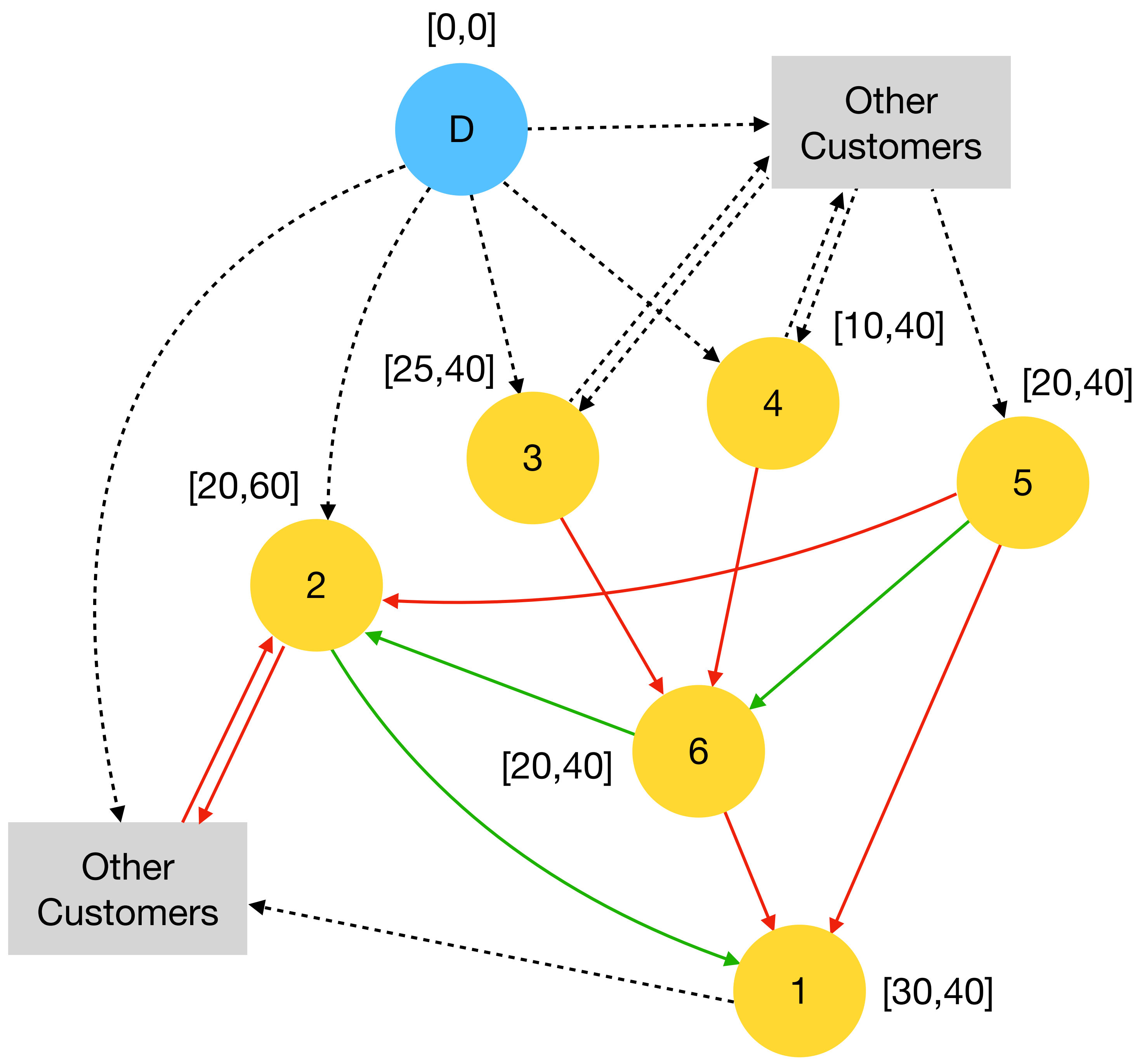
$c_{2,1} = 10$

$t_1 \geq 50$

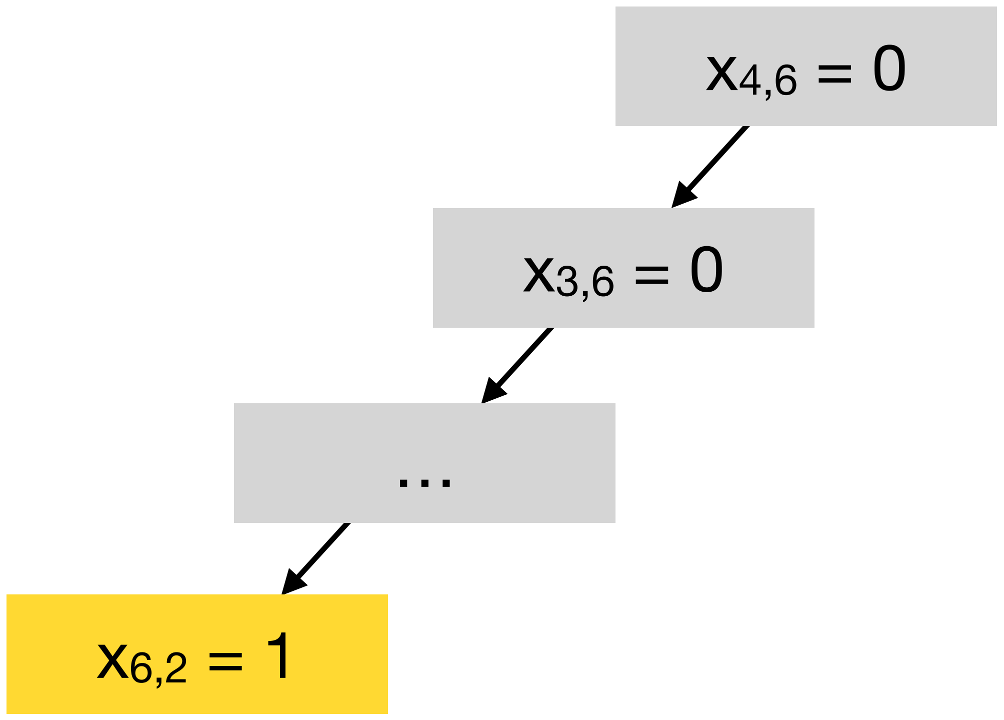
Infeasible

$t_1 \leq 40$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

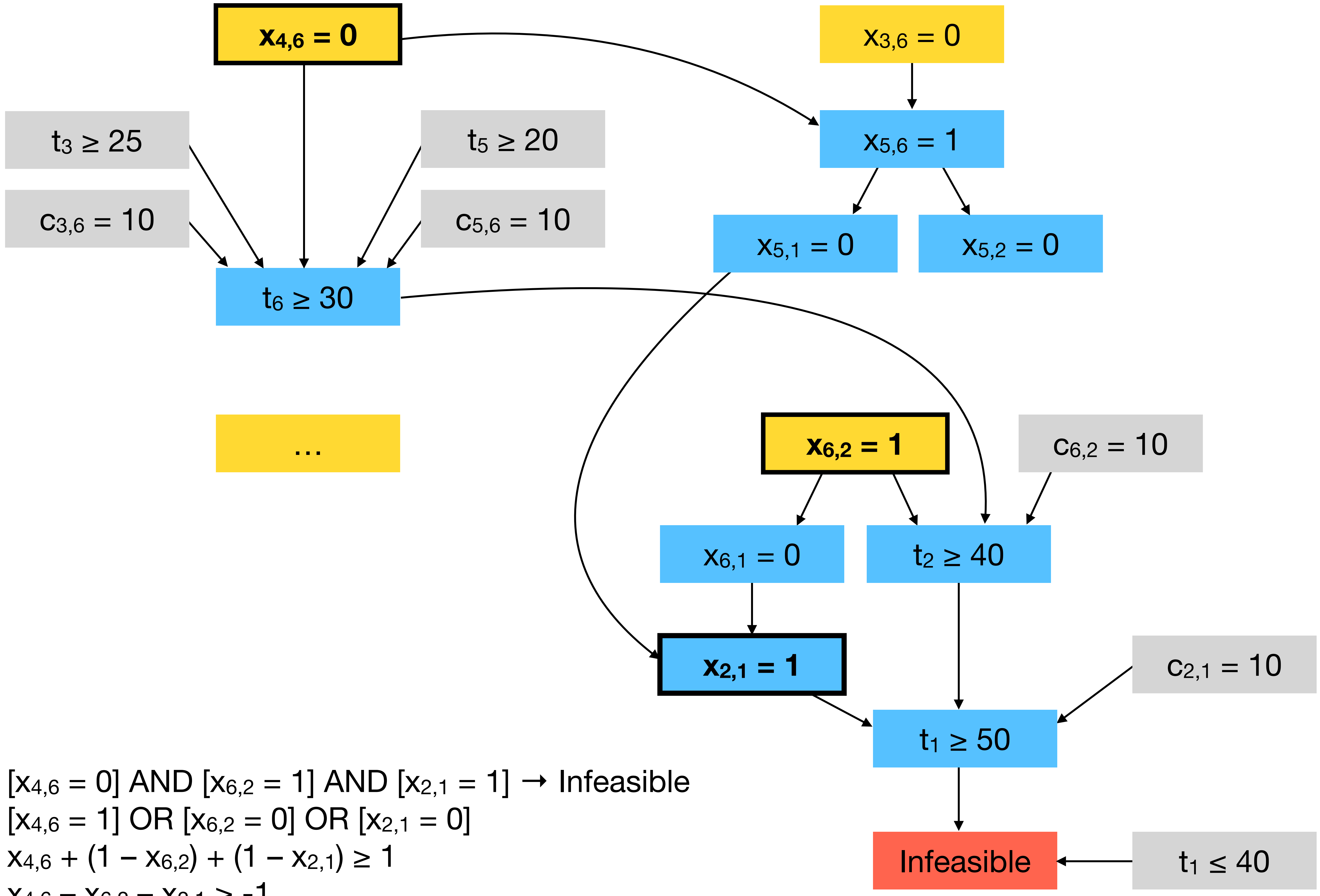
…

$x_{6,2} = 1$

Implication Graph

t_i: earliest time of starting service at vertex i
$c_{i,j}$: cost/travel time from i to j

Data | Decision (branching) | Propagation (inferred)



$x_{4,6} = 0$

$x_{3,6} = 0$

$t_3 \geq 25$

$t_5 \geq 20$

$x_{5,6} = 1$

$c_{3,6} = 10$

$c_{5,6} = 10$

$x_{5,1} = 0$

$x_{5,2} = 0$

$t_6 \geq 30$

…

$x_{6,2} = 1$

$c_{6,2} = 10$

$x_{6,1} = 0$

$t_2 \geq 40$

$x_{2,1} = 1$

$c_{2,1} = 10$

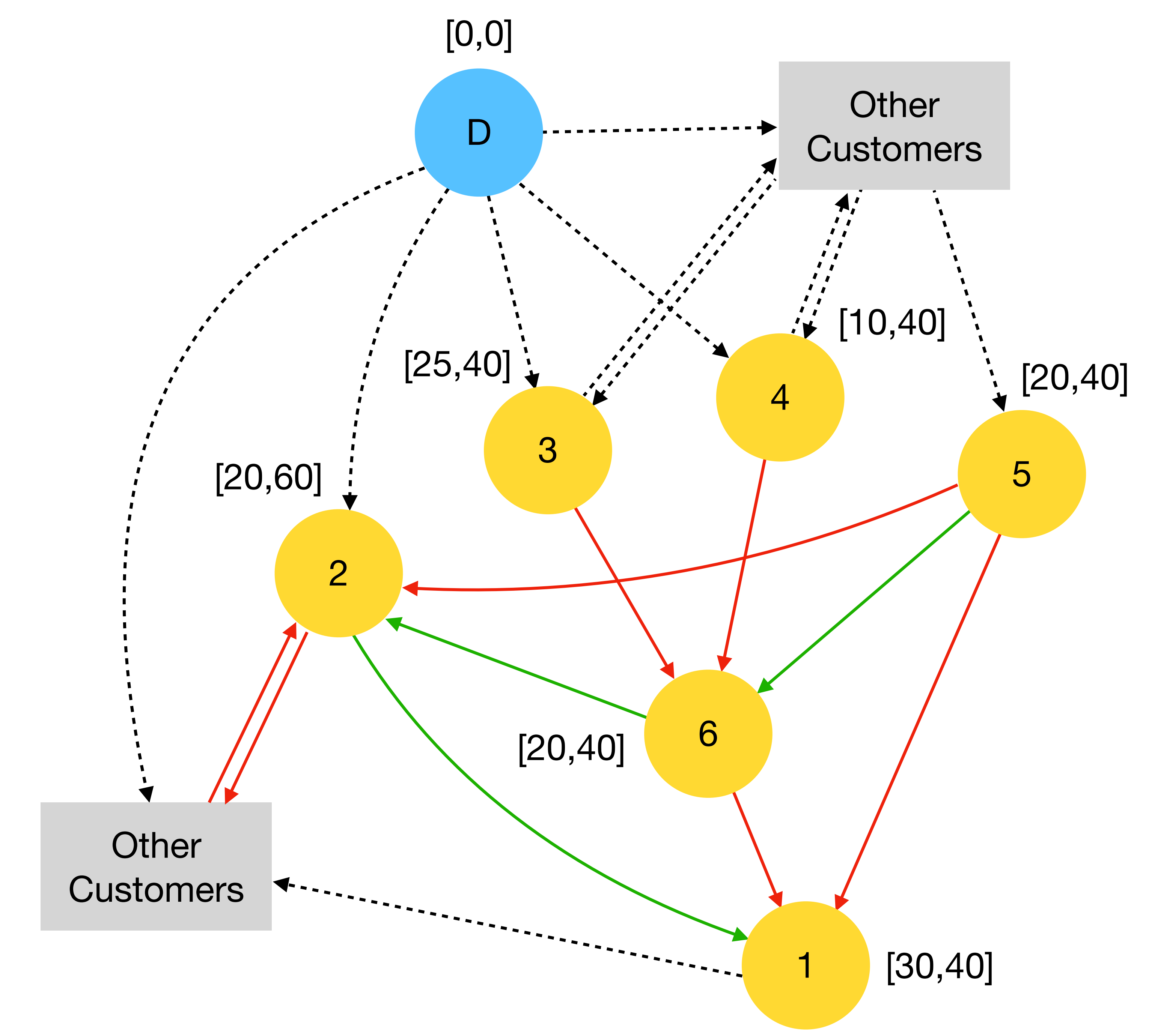
$t_1 \geq 50$

Infeasible

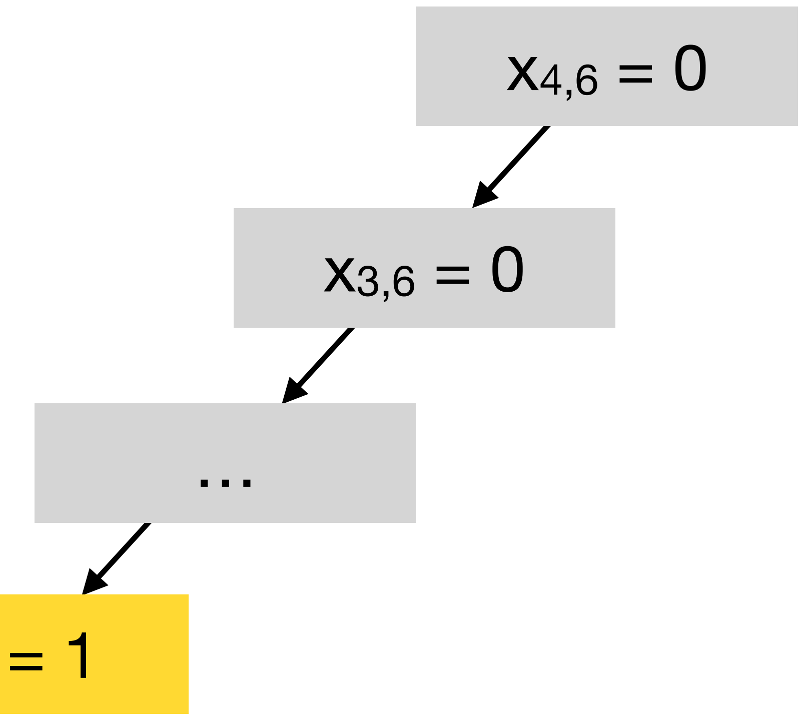
$t_1 \leq 40$

[$x_{4,6} = 0$] AND [$x_{6,2} = 1$] AND [$x_{2,1} = 1$] → Infeasible
[$x_{4,6} = 1$] OR [$x_{6,2} = 0$] OR [$x_{2,1} = 0$]
$x_{4,6} + (1 - x_{6,2}) + (1 - x_{2,1}) \geq 1$
$x_{4,6} - x_{6,2} - x_{2,1} \geq -1$

Network



Branch-and-Bound Tree



$x_{4,6} = 0$

$x_{3,6} = 0$

…

$x_{6,2} = 1$

Caveat

- In the example: trace the propagations backwards in the implication graph until all literals concern variables in the master problem.

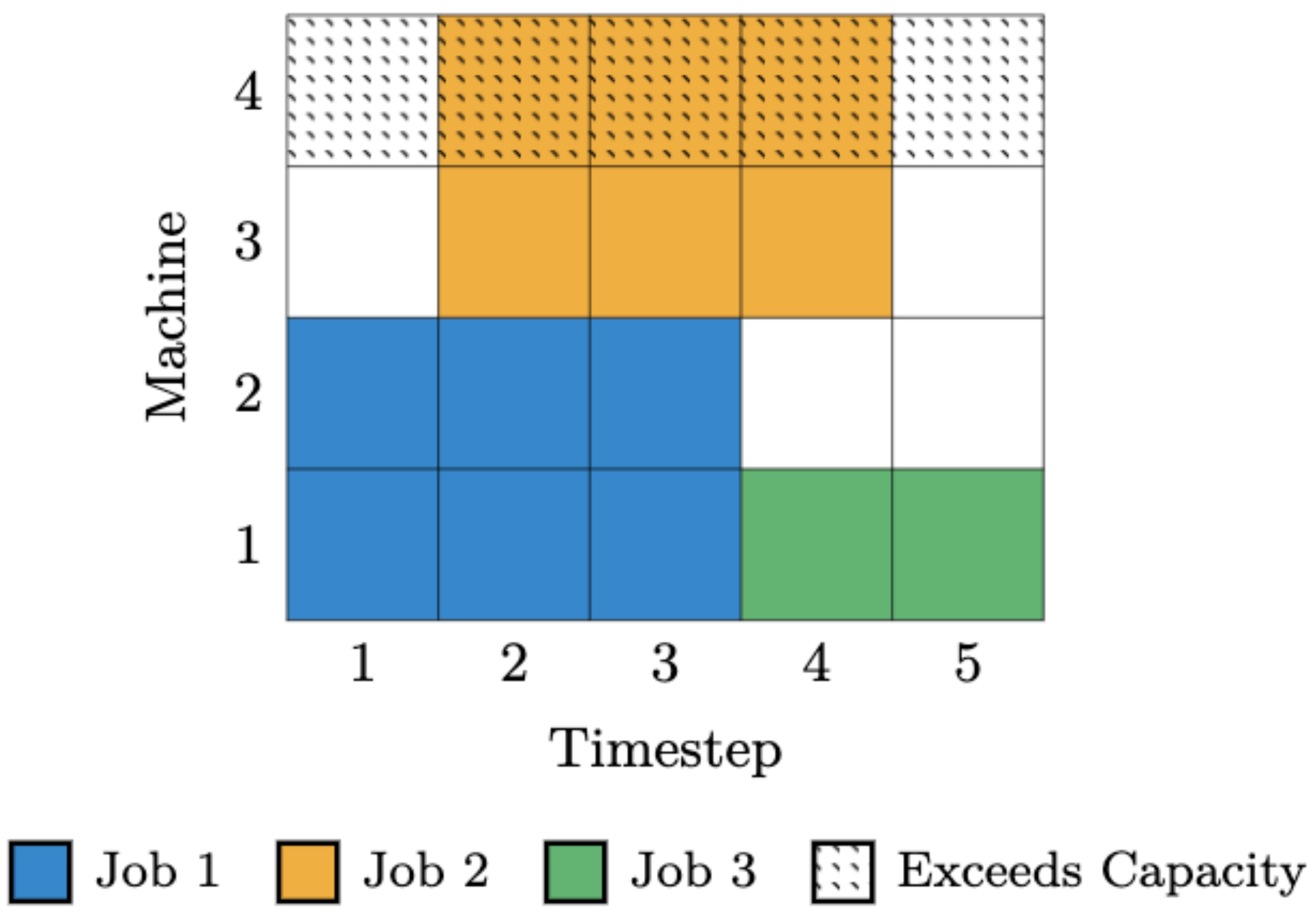
- In practice: trace the implication graph backwards until (1) all literals concern variables in the master problem and (2) there is only one literal at the current depth of the branch-and-bound tree. Get a “first unique implication point” (1UIP) nogood. Experimental evidence from SAT suggests 1UIP nogoods perform better.

Nutmeg

- A proof-of-concept automatic decomposition solver that implements the generic form of logic-based Benders decomposition.
- Calls SCIP for the MIP master problem and branch-and-bound.
- Calls Geas for the CP Benders subproblem and conflict analysis.
- Actually, just a bunch of SCIP plug-ins that glue the master problem and subproblem together.

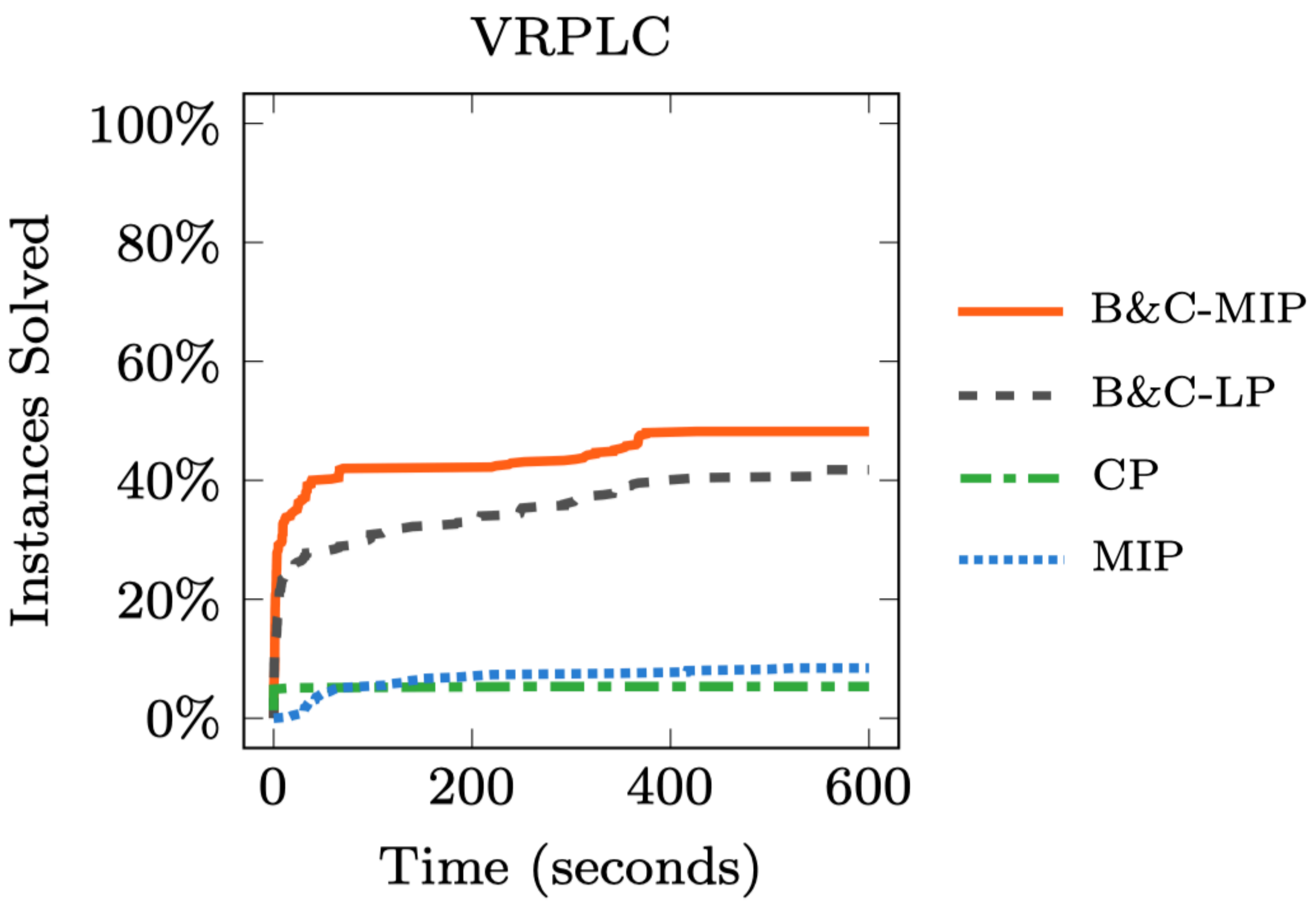
Vehicle Routing Problem with Location Congestion

- Multiple customers have the same (x,y) coordinates. Deliveries to the same warehouse.
- Warehouses have a limited number of machines for unloading the vehicles.
- Vehicles must share the machines to unload.
- VRPLC has joint routing and scheduling combinatorial structure (a type of sychronization).



Experimental Results

- MIP is good at network flow (TSP, VRP).
- CP is good at packing (RCPSP).
- Hybrid appears to be good at vehicle routing with scheduling.



Experimental Results

- Nutmeg tested on a range of problems.
  - LBBD known to perform well on some problems (e.g., VRPLC).
  - LBBD performance unknown on majority of problems. No one has tried due to labor of deriving problem-specific cuts.
- Results confirm LBBD successes without manually deriving cuts.
- Mostly does not work. Problem still needs appropriate structure.
- Discovered one new problem suitable for LBBD.

Key Points

- Logic-based Benders decomposition previously required problem-specific cuts.
- Otherwise, naive combinatorial Benders cuts work but are very weak.
- Conflict analysis can find tighter combinatorial Benders cuts (fewer variables).
- Recently implemented in an automatic decomposition solver named Nutmeg.
- Nutmeg works only on problems with appropriate structure.
- In VRP, “robust” cuts over arcs naturally translate to cuts over paths in branch-and-price.
- VRPs with synchronization become easy/easier. Generate the routes independently and then prevent a subset of arcs that violate the synchronization constraints. But no guarantee it’s fast.
- Super pre-alpha release at https://github.com/ed-lam/nutmeg.