

1 Introduction

This report details the development of the two solution methodologies for the Constrained Shortest Path Problem (CSPP). The first methodology is based on a link/node-based Integer Programming (IP) formulation, employing a general IP solver such as Gurobi or HiGHS. The second methodology is the Branch-and-Price algorithm, which is emphasized more.

No AI tools were utilized for the development and experimental analysis. However, AI assistance was sought for refining the report's language and expression, specifically employing GPT-4.0 for sentence structure and expression enhancement and Grammarly for grammar checks.

Two Julia scripts are submitted:

- **CSPP.jl**: Key functions are defined in this script.
- **main.jl**: This script is the main file performing the experiment by importing functions defined in 'CSPP.jl.' It loads instances, solves them, and saves the results. The results are saved in the 'results' folder as .csv files.

The remainder of this report is organized into three sections. Section 2 introduces the Julia scripts and the methodologies they implement. Section 3 describes the results obtained by the developed codes.

2 Methodologies & codes summary

2.1 Link/node formulation with general IP solver

The Link/node formulation of CSPP is as follows:

$$\text{minimize } \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (1)$$

$$\text{subject to } \sum_{j: (1,j) \in A} x_{1j} = 1, \quad (2)$$

$$\sum_{j: (i,j) \in A} x_{ij} - \sum_{j: (j,i) \in A} x_{ji} = 0, \quad i \in N \setminus \{1, d\} \quad (3)$$

$$\sum_{j: (j,d) \in A} x_{jd} = 1, \quad (4)$$

$$\sum_{(i,j) \in A} c_{ij} x_{ij} \leq T_{\max}, \quad (5)$$

$$x_{ij} \in \{0,1\}, \quad (i,j) \in A \quad (6)$$

where N is the node index set, and A is the arc index set. The decision variable, x_{ij} , indicates whether an arc is used or not. By writing a code throwing the whole formulation to a general IP solver, we can solve CSPP. It is implemented as the function named ‘IPSolver’ as follows:

```

34 function IPSolver(data, Tmax, origin, destination)
35
36 start_node = Int.(data[:, 1])
37 end_node = Int.(data[:, 2])
38 arc_cost = data[:, 3]
39 arc_time = data[:, 4]
40 n_nodes = maximum(vcat(start_node, end_node))
41 n_arcs = length(start_node)
42
43 model = Model(HiGHS.Optimizer)
44
45 @variable(model, x[i = 1:n_arcs], Bin)
46
47 @objective(model, Min, sum(arc_cost .* x))
48
49 @constraint(model, flow_origin, sum(x[start_node .== origin]) - sum(x[end_node .== origin]) == 1)
50 @constraint(model, flow_balance[i = 1:n_nodes; (i != origin) && (i != destination)], sum(x[start_node .== i]) - sum(x[end_node .== i]) == 0)
51 @constraint(model, flow_destination, sum(x[start_node .== destination]) - sum(x[end_node .== destination]) == -1)
52
53 @constraint(model, time, sum(arc_time .* x) <= Tmax)
54
55 optimize!(model)
56
57
58 if primal_status(model) == FEASIBLE_POINT
59     cost_opt, x_opt = objective_value(model), value.(x)
60 else
61     println("Infeasible instance!")
62     cost_opt, x_opt = nothing, nothing
63 end
64
65 path_opt = Set(findall(i->x_opt[i] > 1-EPS, 1:1:n_arcs))
66
67 return cost_opt, path_opt
68
69 end

```

The inputs are ‘data,’ ‘Tmax,’ ‘origin,’ and ‘destination.’ The data argument contains the information of the network as $|N| \times 4$ matrix. Each row represents an arc, and each column represents the tail node index, the head node index, associated cost and time, respectively. The optimal cost, ‘cost_opt,’ and the corresponding path, ‘path_opt’ are provided as outputs.

2.2 Branch-and-Price

Implementation of Branch-and-Price, which is our main purpose, is written in the function named ‘MyCGSolver.’

```

197 function MyCGSolver(data, Tmax, origin, destination)
198
199 start_node = Int.(data[:, 1])
200 end_node = Int.(data[:, 2])
201 arc_cost = data[:, 3]
202 arc_time = data[:, 4]
203 n_nodes = maximum(vcat(start_node, end_node))
204 n_arcs = length(start_node)
205
206 ## initialize
207 # containers for algorithms
208 pred_node = [0]

```

```

209 zero_arcs_node::Vector{Vector{Int}} = [[]]
210 one_arcs_node::Vector{Vector{Int}} = [[]]
211 state_node = [0] # 0: unsolved and not pruned out, 1: integer, 2: fractional, 3: infeasible, 4: pruned out
212 obj_val_node = [0.0]
213 path_ind_node = [Int[]]
214 costs_node = [Number[]]
215 times_node = [Number[]]
216 paths_node = [Set{Int}[]]
217
218 curr_node_ind = 1
219 best_node_ind = 0
220 best_path = Int[]
221 best_obj_val = Inf
222
223 # get an initial feasible solution
224 g = SimpleWeightedDiGraph(n_nodes)
225 for i in 1:n_arcs
226     add_edge!(g, start_node[i], end_node[i], arc_cost[i])
227 end
228 bf_state = bellman_ford_shortest_paths(g, origin)
229 new_path = enumerate_paths(bf_state)[destination]
230 path_arcs = [findfirst(x->(start_node[x]==new_path[i]) && end_node[x]==new_path[i+1], 1:n_arcs) for i in 1:(length(new_path)-1)]
231 tpv = sum(data[path_arcs, 3:4], dims=1)
232 cost, time = tpv[1], tpv[2]
233
234 push!(path_ind_node[curr_node_ind], length(path_ind_node[curr_node_ind])+1)
235 push!(costs_node[curr_node_ind], cost)
236 push!(times_node[curr_node_ind], time)
237 push!(paths_node[curr_node_ind], Set{Int}(path_arcs))
238
239 # algorithm starts
240 while true
241     # solve node by column generation
242     lam_val, obj_val, y0_val, feasible = solve_node!(...)
243
244     # update the current node status & best node
245     if y0_val > EPS # 3: infeasible
246         state_node[curr_node_ind] = 3
247         obj_val_node[curr_node_ind] = Inf
248     elseif sum(lam_val .> 1-EPS) == 1 # 1: integer
249         state_node[curr_node_ind] = 1
250         obj_val_node[curr_node_ind] = obj_val
251         if obj_val < best_obj_val # best solution update
252             best_node_ind = curr_node_ind
253             tpv = findfirst(i->lam_val[i]>1-EPS, path_ind_node[best_node_ind])
254             best_path = paths_node[best_node_ind][tpv]
255             best_obj_val = obj_val
256         end
257     elseif sum(lam_val .> 1-EPS) != 1 # fractional
258         if best_obj_val - obj_val > EPS # 2: not pruned out
259             state_node[curr_node_ind] = 2
260             obj_val_node[curr_node_ind] = obj_val
261             ## branch, strategy: the first fractional arc
262             if feasible
263                 # find the first fractional arc
264                 ...
265             end
266             # branch
267             ...
268
269             # branch
270             ...
271
272             # branch
273             ...
274
275             # branch
276             ...
277
278             # branch
279             ...
280
281             # branch
282             ...
283
284             # branch
285             ...
286
287             # branch
288             ...
289
290             # branch
291             ...
292
293             # branch
294             ...
295
296             # branch
297             ...
298
299             # branch
300             ...
301
302             # branch
303             ...
304
305             # branch
306             ...
307
308             # branch
309             ...
310
311             # branch
312             ...
313
314             # branch
315             ...
316
317             # branch
318             ...
319
320             # branch
321             ...
322
323             # branch
324             ...
325
326             # branch
327             ...
328
329             # branch
330             ...
331
332             # branch
333             ...
334
335             # branch
336             ...
337
338             # branch
339             ...
340
341             # branch
342             ...
343
344             # branch
345             ...
346
347             # branch
348             ...
349
350             # branch
351             ...
352
353             # branch
354             ...
355
356             # branch
357             ...
358
359             # branch
360             ...
361
362             # branch
363             ...
364
365             # branch
366             ...
367
368             # branch
369             ...
370
371             # branch
372             ...
373
374             # branch
375             ...
376
377             # branch
378             ...
379
380             # branch
381             ...
382
383             # branch
384             ...
385
386             # branch
387             ...
388
389             # branch
390             ...
391
392             # branch
393             ...
394
395             # branch
396             ...
397
398             # branch
399             ...
400
401             # branch
402             ...
403
404             # branch
405             ...
406
407             # branch
408             ...
409
410             # branch
411             ...
412
413             # branch
414             ...
415
416             # branch
417             ...
418
419             # branch
420             ...
421
422             # branch
423             ...
424
425             # branch
426             ...
427
428             # branch
429             ...
430
431             # branch
432             ...
433
434             # branch
435             ...
436
437             # branch
438             ...
439
440             # branch
441             ...
442
443             # branch
444             ...
445
446             # branch
447             ...
448
449             # branch
450             ...
451
452             # branch
453             ...
454
455             # branch
456             ...
457
458             # branch
459             ...
460
461             # branch
462             ...
463
464             # branch
465             ...
466
467             # branch
468             ...
469
470             # branch
471             ...
472
473             # branch
474             ...
475
476             # branch
477             ...
478
479             # branch
480             ...
481
482             # branch
483             ...
484
485             # branch
486             ...
487
488             # branch
489             ...
490
491             # branch
492             ...
493
494             # branch
495             ...
496
497             # branch
498             ...
499
500             # branch
501             ...
502
503             # branch
504             ...
505
506             # branch
507             ...
508
509             # branch
510             ...
511
512             # branch
513             ...
514
515             # branch
516             ...
517
518             # branch
519             ...
520
521             # branch
522             ...
523
524             # branch
525             ...
526
527             # branch
528             ...
529
530             # branch
531             ...
532
533             # branch
534             ...
535
536             # branch
537             ...
538
539             # branch
540             ...
541
542             # branch
543             ...
544
545             # branch
546             ...
547
548             # branch
549             ...
550
551             # branch
552             ...
553
554             # branch
555             ...
556
557             # branch
558             ...
559
560             # branch
561             ...
562
563             # branch
564             ...
565
566             # branch
567             ...
568
569             # branch
570             ...
571
572             # branch
573             ...
574
575             # branch
576             ...
577
578             # branch
579             ...
580
581             # branch
582             ...
583
584             # branch
585             ...
586
587             # branch
588             ...
589
590             # branch
591             ...
592
593             # branch
594             ...
595
596             # branch
597             ...
598
599             # branch
600             ...
601
602             # branch
603             ...
604
605             # branch
606             ...
607
608             # branch
609             ...
610
611             # branch
612             ...
613
614             # branch
615             ...
616
617             # branch
618             ...
619
620             # branch
621             ...
622
623             # branch
624             ...
625
626             # branch
627             ...
628
629             # branch
630             ...
631
632             # branch
633             ...
634
635             # branch
636             ...
637
638             # branch
639             ...
640
641             # branch
642             ...
643
644             # branch
645             ...
646
647             # branch
648             ...
649
650             # branch
651             ...
652
653             # branch
654             ...
655
656             # branch
657             ...
658
659             # branch
660             ...
661
662             # branch
663             ...
664
665             # branch
666             ...
667
668             # branch
669             ...
670
671             # branch
672             ...
673
674             # branch
675             ...
676
677             # branch
678             ...
679
680             # branch
681             ...
682
683             # branch
684             ...
685
686             # branch
687             ...
688
689             # branch
690             ...
691
692             # branch
693             ...
694
695             # branch
696             ...
697
698             # branch
699             ...
700
701             # branch
702             ...
703
704             # branch
705             ...
706
707             # branch
708             ...
709
710             # branch
711             ...
712
713             # branch
714             ...
715
716             # branch
717             ...
718
719             # branch
720             ...
721
722             # branch
723             ...
724
725             # branch
726             ...
727
728             # branch
729             ...
730
731             # branch
732             ...
733
734             # branch
735             ...
736
737             # branch
738             ...
739
740             # branch
741             ...
742
743             # branch
744             ...
745
746             # branch
747             ...
748
749             # branch
750             ...
751
752             # branch
753             ...
754
755             # branch
756             ...
757
758             # branch
759             ...
760
761             # branch
762             ...
763
764             # branch
765             ...
766
767             # branch
768             ...
769
770             # branch
771             ...
772
773             # branch
774             ...
775
776             # branch
777             ...
778
779             # branch
780             ...
781
782             # branch
783             ...
784
785             # branch
786             ...
787
788             # branch
789             ...
790
791             # branch
792             ...
793
794             # branch
795             ...
796
797             # branch
798             ...
799
800             # branch
801             ...
802
803             # branch
804             ...
805
806             # branch
807             ...
808
809             # branch
810             ...
811
812             # branch
813             ...
814
815             # branch
816             ...
817
818             # branch
819             ...
820
821             # branch
822             ...
823
824             # branch
825             ...
826
827             # branch
828             ...
829
830             # branch
831             ...
832
833             # branch
834             ...
835
836             # branch
837             ...
838
839             # branch
840             ...
841
842             # branch
843             ...
844
845             # branch
846             ...
847
848             # branch
849             ...
850
851             # branch
852             ...
853
854             # branch
855             ...
856
857             # branch
858             ...
859
860             # branch
861             ...
862
863             # branch
864             ...
865
866             # branch
867             ...
868
869             # branch
870             ...
871
872             # branch
873             ...
874
875             # branch
876             ...
877
878             # branch
879             ...
880
881             # branch
882             ...
883
884             # branch
885             ...
886
887             # branch
888             ...
889
890             # branch
891             ...
892
893             # branch
894             ...
895
896             # branch
897             ...
898
899             # branch
900             ...
901
902             # branch
903             ...
904
905             # branch
906             ...
907
908             # branch
909             ...
910
911             # branch
912             ...
913
914             # branch
915             ...
916
917             # branch
918             ...
919
920             # branch
921             ...
922
923             # branch
924             ...
925
926             # branch
927             ...
928
929             # branch
930             ...
931
932             # branch
933             ...
934
935             # branch
936             ...
937
938             # branch
939             ...
940
941             # branch
942             ...
943
944             # branch
945             ...
946
947             # branch
948             ...
949
950             # branch
951             ...
952
953             # branch
954             ...
955
956             # branch
957             ...
958
959             # branch
960             ...
961
962             # branch
963             ...
964
965             # branch
966             ...
967
968             # branch
969             ...
970
971             # branch
972             ...
973
974             # branch
975             ...
976
977             # branch
978             ...
979
980             # branch
981             ...
982
983             # branch
984             ...
985
986             # branch
987             ...
988
989             # branch
990             ...
991
992             # branch
993             ...
994
995             # branch
996             ...
997
998             # branch
999             ...
1000             # branch
1001             ...
1002             # branch
1003             ...
1004             # branch
1005             ...
1006             # branch
1007             ...
1008             # branch
1009             ...
1010             # branch
1011             ...
1012             # branch
1013             ...
1014             # branch
1015             ...
1016             # branch
1017             ...
1018             # branch
1019             ...
1020             # branch
1021             ...
1022             # branch
1023             ...
1024             # branch
1025             ...
1026             # branch
1027             ...
1028             # branch
1029             ...
1030             # branch
1031             ...
1032             # branch
1033             ...
1034             # branch
1035             ...
1036             # branch
1037             ...
1038             # branch
1039             ...
1040             # branch
1041             ...
1042             # branch
1043             ...
1044             # branch
1045             ...
1046             # branch
1047             ...
1048             # branch
1049             ...
1050             # branch
1051             ...
1052             # branch
1053             ...
1054             # branch
1055             ...
1056             # branch
1057             ...
1058             # branch
1059             ...
1060             # branch
1061             ...
1062             # branch
1063             ...
1064             # branch
1065             ...
1066             # branch
1067             ...
1068             # branch
1069             ...
1070             # branch
1071             ...
1072             # branch
1073             ...
1074             # branch
1075             ...
1076             # branch
1077             ...
1078             # branch
1079             ...
1080             # branch
1081             ...
1082             # branch
1083             ...
1084             # branch
1085             ...
1086             # branch
1087             ...
1088             # branch
1089             ...
1090             # branch
1091             ...
1092             # branch
1093             ...
1094             # branch
1095             ...
1096             # branch
1097             ...
1098             # branch
1099             ...
1100             # branch
1101             ...
1102             # branch
1103             ...
1104             # branch
1105             ...
1106             # branch
1107             ...
1108             # branch
1109             ...
1110             # branch
1111             ...
1112             # branch
1113             ...
1114             # branch
1115             ...
1116             # branch
1117             ...
1118             # branch
1119             ...
1120             # branch
1121             ...
1122             # branch
1123             ...
1124             # branch
1125             ...
1126             # branch
1127             ...
1128             # branch
1129             ...
1130             # branch
1131             ...
1132             # branch
1133             ...
1134             # branch
1135             ...
1136             # branch
1137             ...
1138             # branch
1139             ...
1140             # branch
1141             ...
1142             # branch
1143             ...
1144             # branch
1145             ...
1146             # branch
1147             ...
1148             # branch
1149             ...
1150             # branch
1151             ...
1152             # branch
1153             ...
1154             # branch
1155             ...
1156             # branch
1157             ...
1158             # branch
1159             ...
1160             # branch
1161             ...
1162             # branch
1163             ...
1164             # branch
1165             ...
1166             # branch
1167             ...
1168             # branch
1169             ...
1170             # branch
1171             ...
1172             # branch
1173             ...
1174             # branch
1175             ...
1176             # branch
1177             ...
1178             # branch
1179             ...
1180             # branch
1181             ...
1182             # branch
1183             ...
1184             # branch
1185             ...
1186             # branch
1187             ...
1188             # branch
1189             ...
1190             # branch
1191             ...
1192             # branch
1193             ...
1194             # branch
1195             ...
1196             # branch
1197             ...
1198             # branch
1199             ...
1200             # branch
1201             ...
1202             # branch
1203             ...
1204             # branch
1205             ...
1206             # branch
1207             ...
1208             # branch
1209             ...
1210             # branch
1211             ...
1212             # branch
1213             ...
1214             # branch
1215             ...
1216             # branch
1217             ...
1218             # branch
1219             ...
1220             # branch
1221             ...
1222             # branch
1223             ...
1224             # branch
1225             ...
1226             # branch
1227             ...
1228             # branch
1229             ...
1230             # branch
1231             ...
1232             # branch
1233             ...
1234             # branch
1235             ...
1236             # branch
1237             ...
1238             # branch
1239             ...
1240             # branch
1241             ...
1242             # branch
1243             ...
1244             # branch
1245             ...
1246             # branch
1247             ...
1248             # branch
1249             ...
1250             # branch
1251             ...
1252             # branch
1253             ...
1254             # branch
1255             ...
1256             # branch
1257             ...
1258             # branch
1259             ...
1260             # branch
1261             ...
1262             # branch
1263             ...
1264             # branch
1265             ...
1266             # branch
1267             ...
1268             # branch
1269             ...
1270             # branch
1271             ...
1272             # branch
1273             ...
1274             # branch
1275             ...
1276             # branch
1277             ...
1278             # branch
1279             ...
1280             # branch
1281             ...
1282             # branch
1283             ...
1284             # branch
1285             ...
1286             # branch
1287             ...
1288             # branch
1289             ...
1290             # branch
1291             ...
1292             # branch
1293             ...
1294             # branch
1295             ...
1296             # branch
1297             ...
1298             # branch
1299             ...
1300             # branch
1301             ...
1302             # branch
1303             ...
1304             # branch
1305             ...
1306             # branch
1307             ...
1308             # branch
1309             ...
1310             # branch
1311             ...
1312             # branch
1313             ...
1314             # branch
1315             ...
1316             # branch
1317             ...
1318             # branch
1319             ...
1320             # branch
1321             ...
1322             # branch
1323             ...
1324             # branch
1325             ...
1326             # branch
1327             ...
1328             # branch
1329             ...
1330             # branch
1331             ...
1332             # branch
1333             ...
1334             # branch
1335             ...
1336             # branch
1337             ...
1338             # branch
1339             ...
1340             # branch
1341             ...
1342             # branch
1343             ...
1344             # branch
1345             ...
1346             # branch
1347             ...
1348             # branch
1349             ...
1350             # branch
1351             ...
1352             # branch
1353             ...
1354             # branch
1355             ...
1356             # branch
1357             ...
1358             # branch
1359             ...
1360             # branch
1361             ...
1362             # branch
1363             ...
1364             # branch
1365             ...
1366             # branch
1367             ...
1368             # branch
1369             ...
1370             # branch
1371             ...
1372             # branch
1373             ...
1374             # branch
1375             ...
1376             # branch
1377             ...
1378             # branch
1379             ...
1380             # branch
1381             ...
1382             # branch
1383             ...
1384             # branch
1385             ...
1386             # branch
1387             ...
1388             # branch
1389             ...
1390             # branch
1391             ...
1392             # branch
1393             ...
1394             # branch
1395             ...
1396             # branch
1397             ...
1398             # branch
1399             ...
1400             # branch
1401             ...
1402             # branch
1403             ...
1404             # branch
1405             ...
1406             # branch
1407             ...
1408             # branch
1409             ...
1410             # branch
1411             ...
1412             # branch
1413             ...
1414             # branch
1415             ...
1416             # branch
1417             ...
1418             # branch
1419             ...
1420             # branch
1421             ...
1422             # branch
1423             ...
1424             # branch
1425             ...
1426             # branch
1427             ...
1428             # branch
1429             ...
1430             # branch
1431             ...
1432             # branch
1433             ...
1434             # branch
1435             ...
1436             # branch
1437             ...
1438             # branch
1439             ...
1440             # branch
1441             ...
1442             # branch
1443             ...
1444             # branch
1445             ...
1446             # branch
1447             ...
1448             # branch
1449             ...
1450             # branch
1451             ...
1452             # branch
1453             ...
1454             # branch
1455             ...
1456             # branch
1457             ...
1458             # branch
1459             ...
1460             # branch
1461             ...
1462             # branch
1463             ...
1464             # branch
1465             ...
1466             # branch
1467             ...
1468             # branch
1469             ...
1470             # branch
1471             ...
1472             # branch
1473             ...
1474             # branch
1475             ...
1476             # branch
1477             ...
1478             # branch
1479             ...
1480             # branch
1481             ...
1482             # branch
1483             ...
1484             # branch
1485             ...
1486             # branch
1487             ...
1488             # branch
1489             ...
1490             # branch
1491             ...
1492             # branch
1493             ...
1494             # branch
1495             ...
1496             # branch
1497             ...
1498             # branch
1499             ...
1500             # branch
1501             ...
1502             # branch
1503             ...
1504             # branch
1505             ...
1506             # branch
1507             ...
1508             # branch
1509             ...
1510             # branch
1511             ...
1512             # branch
1513             ...
1514             # branch
1515             ...
1516             # branch
1517             ...
1518             # branch
1519             ...
1520             # branch
1521             ...
1522             # branch
1523             ...
1524             # branch
1525             ...
1526             # branch
1527             ...
1528             # branch
1529             ...
1530             # branch
1531             ...
1532             # branch
1533             ...
1534             # branch
1535             ...
1536             # branch
1537             ...
1538             # branch
1539             ...
1540             # branch
1541             ...
1542             # branch
1543             ...
1544             # branch
1545             ...
1546             # branch
1547             ...
1548             # branch
1549             ...
1550             # branch
1551             ...
1552             # branch
1553             ...
1554             # branch
1555             ...
1556             # branch
1557             ...
1558             # branch
1559             ...
1560             # branch
1561             ...
1562             # branch
1563             ...
1564             # branch
1565             ...
1566             # branch
1567             ...
1568             # branch
1569             ...
1570             # branch
1571             ...
1572             # branch
1573             ...
1574             # branch
1575             ...
1576             # branch
1577             ...
1578             # branch
1579             ...
1580             # branch
1581             ...
1582             # branch
1583             ...
1584             # branch
1585             ...
1586             # branch
1587             ...
1588             # branch
1589             ...
1590             # branch
1591             ...
1592             # branch
1593             ...
1594             # branch
1595             ...
1596             # branch
1597             ...
1598             # branch
1599             ...
1600             # branch
1601             ...
1602             # branch
1603             ...
1604             # branch
1605             ...
1606             # branch
1607             ...
1608             # branch
1609             ...
1610             # branch
1611             ...
1612             # branch
1613             ...
1614             # branch
1615             ...
1616             # branch
1617             ...
1618             # branch
1619             ...
1620             # branch
1621             ...
1622             # branch
1623             ...
1624             # branch
1625             ...
1626             # branch
1627             ...
1628             # branch
1629             ...
1630             # branch
1631             ...
1632             # branch
1633             ...
1634             # branch
1635             ...
1636             # branch
1637             ...
1638             # branch
1639             ...
1640             # branch
1641             ...
1642             # branch
1643             ...
1644             # branch
1645             ...
1646             # branch
1647             ...
1648             # branch
1649             ...
1650             # branch
1651             ...
1652             # branch
1653             ...
1654             # branch
1655             ...
1656             # branch
1657             ...
1658             # branch
1659             ...
1660             # branch
1661             ...
1662             # branch
1663             ...
1664             # branch
1665             ...
1666             # branch
1667             ...
1668             # branch
1669             ...
1670             # branch
1671             ...
1672             # branch
1673             ...
1674             # branch
1675             ...
1676             # branch
1677             ...
1678             # branch
1679             ...
1680             # branch
1681             ...
1682             # branch
1683             ...
1684             # branch
1685             ...
1686             # branch
1687             ...
1688             # branch
1689             ...
1690             # branch
1691             ...
1692             # branch
1693             ...
1694             # branch
1695             ...
1696             # branch
1697             ...
1698             # branch
1699             ...
1700             # branch
1701             ...
1702             # branch
1703             ...
1704             # branch
1705             ...
1706             # branch
1707             ...
1708             # branch
1709             ...
1710             # branch
1711             ...
1712             # branch
1713             ...
1714             # branch
1715             ...
1716             # branch
1717             ...
1718             # branch
1719             ...
1720             # branch
1721             ...
1722             # branch
1723             ...
1724             # branch
1725             ...
1726             # branch
1727             ...
1728             # branch
1729             ...
1730             # branch
1731             ...
1732             # branch
1733             ...
1734             # branch
1735             ...
1736             # branch
1737             ...
1738             # branch
1739             ...
1740             # branch
1741             ...
1742             # branch
1743             ...
1744             # branch
1745             ...
1746             # branch
1747             ...
1748             # branch
1749             ...
1750             # branch
1751             ...
1752             # branch
1753             ...
1754             # branch
1755             ...
1756             # branch
1757             ...
1758             # branch
1759             ...
1760             # branch
1761             ...
1762             # branch
1763             ...
1764             # branch
1765             ...
1766             # branch
1767             ...
1768             # branch
1769             ...
1770             # branch
1771             ...
1772             # branch
1773             ...
1774             # branch
1775             ...
1776             # branch
1777             ...
1778             # branch
1779             ...
1780             # branch
1781             ...
1782             # branch
1783             ...
1784             # branch
1785             ...
1786             # branch
1787             ...
1788             # branch
1789             ...
1790             # branch
1791             ...
1792             # branch
1793             ...
1794             # branch
1795             ...
1796             # branch
1797             ...
1798             # branch
1799             ...
1800             # branch
1801             ...
1802             # branch
1803             ...
1804             # branch
1805             ...
1806             # branch
1807             ...
1808             # branch
1809             ...
1810             # branch
1811             ...
1812             # branch
1813             ...
1814             # branch
1815             ...
1816             # branch
1817             ...
1818             # branch
1819             ...
1820             # branch
1821             ...
1822             # branch
1823             ...
1824             # branch
1825             ...
1826             # branch
1827             ...
1828             # branch
1829             ...
1830             # branch
1831             ...
1832             # branch
1833             ...
1834             # branch
1835             ...
1836             # branch
1837             ...
1838             # branch
1839             ...
1840             # branch
1841             ...
1842             # branch
1843             ...
1844             # branch
1845             ...
1846             # branch
1847             ...
1848             # branch
1849             ...
1850             # branch
1851             ...
1852             # branch
1853             ...
1854             # branch
1855             ...
1856             # branch
1857             ...
1858             # branch
1859             ...
1860             # branch
1861             ...
1862             # branch
1863             ...
1864             # branch
1865             ...
1866             # branch
1867             ...
1868             # branch
1869             ...
1870             # branch
1871             ...
1872             # branch
1873             ...
1874             # branch
1875             ...
1876             # branch
1877             ...
1878             # branch
1879             ...
1880             # branch
1881             ...
1882             # branch
1883             ...
1884             # branch
1885             ...
1886             # branch
1887             ...
1888             # branch
1889             ...
1890             # branch
1891             ...
1892             # branch
1893             ...
1894             # branch
1895             ...
1896             # branch
1897             ...
1898             # branch
1899             ...
1900             # branch
1901             ...
1902             # branch
1903             ...
1904             # branch
1905             ...
1906             # branch
1907             ...
1908             # branch
1909             ...
1910             # branch
1911             ...
1912             # branch
1913             ...
1914             # branch
1915             ...
1916             # branch
1917             ...
1918             # branch
1919             ...
1920             # branch
1921             ...
1922             # branch
1923             ...
1924             # branch
1925             ...
1926             # branch
1927             ...
1928             # branch
1929             ...
1930             # branch
1931             ...
1932             # branch
1933             ...
1934             # branch
1935             ...
1936             # branch
1937             ...
1938             # branch
1939             ...
1940             # branch
1941             ...
1942             # branch
1943             ...
1944             # branch
1945             ...
1946             # branch
1947             ...
1948             # branch
1949             ...
1950             # branch
1951             ...
1952             # branch
1953             ...
1954             # branch
1955             ...
1956             # branch
1957             ...
1958             # branch
1959             ...
1960             # branch
1961             ...
1962             # branch
1963             ...
1964             # branch
1965             ...
1966             # branch
1967             ...
1968             # branch
1969             ...
1970             # branch
1971             ...
1972             # branch
1973             ...
1974             # branch
1975             ...
1976             # branch
1977             ...
1978             # branch
1979            
```

```

290     for bInd in eachindex(state_node)
291         if state_node[bInd] == 0
292             curr_node_ind = bInd
293             break
294         end
295     end
296
297     if curr_node_ind == 0      # if no node to solve, terminate the algorithm
298         break
299     end
300 end
301
302 # solution history
303 solution_history = (pred_node, one_arcs_node, zero_arcs_node, state_node, obj_val_node)
304 best_solution = (best_node_ind, one_arcs_node[best_node_ind], zero_arcs_node[best_node_ind], best_obj_val, best_path)
305
306 return solution_history, best_solution
307
308 end

```

The inputs are the same as the IPSolver function. The overall process consists of 3 parts: 1) initialization (lines 206 to 237), 2) solve node (line 242), and 3) branch (lines 244 to 299).

In the initialization phase, the containers for the algorithm are defined, and an initial path to be included in the Restricted Master Problem (RMP) is obtained to start the algorithm. The containers are as follows:

- pred_node: the predecessor node index
- zero_arcs_node: the arc index that is constrained to be 0, by branching
- one_arcs_node: the arc index that is constrained to be 1, by branching
- state_node: state of nodes (0 - unsolved and not pruned out, 1 - integer solution, 2 - fractional solution, 3 - infeasible, 4 - pruned out)
- obj_val_node: the obtained optimal solution of each node
- path_ind_node: paths index inherited from the predecessor node and obtained from solving pricing sub-problems
- costs_node: costs of paths in path_ind_node
- times_node: times of paths in path_ind_node
- paths_node: selected arcs set of paths in path_ind_node

As the Branch-and-Price algorithm progresses, the containers are updated to keep the information of each node. The initial path is obtained using the Bellman-Ford algorithm, which does not guarantee the feasibility of the path in the CSPP; however, starting from this solution is acceptable.

The node solution procedure is written as the supporting function named 'solve_node!'

```

72 function solve_node!(...)
73
74     ## select paths satisfying the node conditions
75     path_ind_rmp = Int[]
76
77     if isempty(one_arcs)
78         path_ind_rmp = copy(path_ind)
79     else

```

```

80     for pInd in path_ind, arc in paths[pInd]
81         if (arc in one_arcs) && !(pInd in path_ind_rmp)
82             push!(path_ind_rmp, pInd)
83         end
84     end
85 end
86
87 # delete zero arcs
88 if isempty(zero_arcs)
89     for pInd in path_ind_rmp, arc in paths[pInd]
90         if (arc in zero_arcs) && (pInd in path_ind_rmp)
91             deleteat!(path_ind_rmp, findfirst(x->x==pInd, path_ind_rmp))
92         end
93     end
94 end
95
96 M = 100.0          # Big-M
97 while true
98     # restricted master problem
99     rmp = Model(HiGHS.Optimizer)
100
101     @variable(rmp, y0 >= 0)                # artificial variable facilitating the solution procedure
102     @variable(rmp, lam[path_ind_rmp] >= 0) # path variables
103     @objective(rmp, Min, M*y0 + sum(costs[path_ind_rmp] .* lam)) # cost minimization
104     @constraint(rmp, resource, sum(times[path_ind_rmp] .* lam) <= Tmax) # resource constraint, 1
105     @constraint(rmp, convexity, y0 + sum(lam) == 1) # convexity constraint, 0
106     optimize!(rmp) # solve the relaxed RMP
107
108     # results
109     y0_val = value(y0)
110     lam_val = value(lam)
111     obj_val = objective_value(rmp)
112     pi0 = dual(convexity)
113     pi1 = dual(resource)
114
115     # if infeasible
116     if y0_val > EPS
117         pi0 = M
118     end
119
120     # pricing sub-problem
121     psp = Model(HiGHS.Optimizer)
122     @variable(psp, x[i = 1:n_arcs], Bin)
123     @objective(psp, Min, sum((arc_cost.-arc_time.*pi1) .* x))
124     @constraint(psp, flow_origin, sum(x[start_node .== origin] - sum(x[end_node .== origin]) == 1)
125     @constraint(psp, flow_balance[i = 1:n_nodes; (i != origin) && (i != destination)], sum(x[start_node .== i]) - sum(x[end_node .== i]) == 0)
126     @constraint(psp, flow_destination, sum(x[start_node .== destination]) - sum(x[end_node .== destination]) == -1)
127     if !isempty(one_arcs)
128         @constraint(psp, node_ones[i = one_arcs], x[i] == 1)
129     end
130     if !isempty(zero_arcs)
131         @constraint(psp, node_zeros[i = zero_arcs], x[i] == 0)
132     end
133     @constraint(psp, time, sum(arc_time .* x) <= Tmax)
134     optimize!(psp)
135
136     # no feasible path within node
137     if primal_status(psp) != FEASIBLE_POINT
138         lam_val_full = zeros(Float64, length(path_ind))
139         lam_val_full[path_ind_rmp] = lam_val.data
140         return lam_val_full, obj_val, y0_val, false
141     end
142
143     reduced_cost = objective_value(psp) - pi0
144
145     if y0_val > EPS # RMP is infeasible
146         x_opt = value(x)
147         new_path = Set(findall(i->x_opt[i]>(1-EPS), collect(1:1:n_arcs)))
148         if new_path in paths[path_ind_rmp] # no column to generate
149             lam_val_full = zeros(Float64, length(path_ind))
150             lam_val_full[path_ind_rmp] = lam_val.data
151             return lam_val_full, obj_val, y0_val, true
152         elseif reduced_cost < -EPS
153             x_opt = value(x)
154             new_path = Set(findall(i->x_opt[i]>(1-EPS), collect(1:1:n_arcs)))
155             new_cost = sum(arc_cost[collect(new_path)])
156             new_time = sum(arc_time[collect(new_path)])
157             # add column to the RMP
158             push!(path_ind, length(path_ind)+1)
159             push!(costs, new_cost)
160             push!(times, new_time)

```

```

161         push!(paths, new_path)
162         push!(path_ind_rmp, length(path_ind))
163     else
164         M *= 10.0
165     end
166 else # RMP is feasible
167     if reduced_cost < -EPS # find a new column
168         x_opt = value.(x)
169         new_path = Set(findall(i->x_opt[i]>(1-EPS), collect(1:n_arcs)))
170         new_cost = sum(arc_cost[collect(new_path)])
171         new_time = sum(arc_time[collect(new_path)])
172         # add column to the RMP
173         push!(path_ind, length(path_ind)+1)
174         push!(costs, new_cost)
175         push!(times, new_time)
176         push!(paths, new_path)
177         push!(path_ind_rmp, length(path_ind))
178     else # no column to generate
179         lam_val_full = zeros{Float64, length(path_ind)}
180         lam_val_full[path_ind_rmp] = lam_val.data
181         return lam_val_full, obj_val, y0_val, true
182     end
183 end
184 end
185 end
186 end

```

The inputs of this function are omitted in this report due to its extensive length of codes. In this function, the RMP and pricing sub-problem are solved iteratively. To keep generating columns even if the RMP is infeasible with the current columns, an artificial decision variable ‘y0’ is introduced in the RMP.

This process has two terminal conditions: First, it terminates if no feasible path satisfies the branching constraints. Specifically, if branching mandates that certain x_{ij} should be kept 0 or 1, and no path satisfies all those conditions, then the node solution process terminates. Second, the process also terminates if there is no column with a negative reduced cost.

Finally, after solving a node, the branch process is initiated. Each node is labeled following the results obtained by the solution process. If the solution is infeasible, integer, or dominated by the current best solution, the node does not branch. However, if the solution is fractional and not dominated by the best solution, the node does branch. To branch out, the first fractional arc in the data that has not been selected to branch is chosen. Two branches are generated, whose selected arc values are 0 and 1, respectively. The details of the codes (lines 263 to 280) can be found in the Julia script.

After the branching process is over, the algorithm finds the next node to solve. If no node remains unsolved, the whole algorithm terminates.

3 Results: test problems

Two given problems in the assignment guide were solved using the IPSolver and the MyCGSolver functions. Both solvers provided the same optimal solution for each problem. Since the Branch-and-Price algorithm is our main target, this section will show its results.

The experiments were performed on my personal computer located in my laboratory, N7-2 3332. The specifications of the computer, language, and used packages are in Table 1.

Table 1: Experimental Environment

Resource	Specification
CPU	13th Gen Intel(R) Core(TM) i9-13900K
RAM	128GB
OS	Windows 11
Language	Julia 1.10.0
Used packages	Graphs (v1.9.0) HiGHS (v1.9.0) JuMP (v1.20.0) JLD (v0.13.5) SimpleWeightedGraphs (v1.4.0)

The results of the first problem are provided as Figure 1. The root node (Node 1) yielded a fractional solution. Following the branching strategy outlined in the previous section, the first fractional arc, (1,2), was chosen for branching. Both Nodes 2 and 3 resulted in integer solutions, eliminating the need for further branching. Node 3 presented the optimal solution with a cost of 13 and the path 1->3->2->4->6

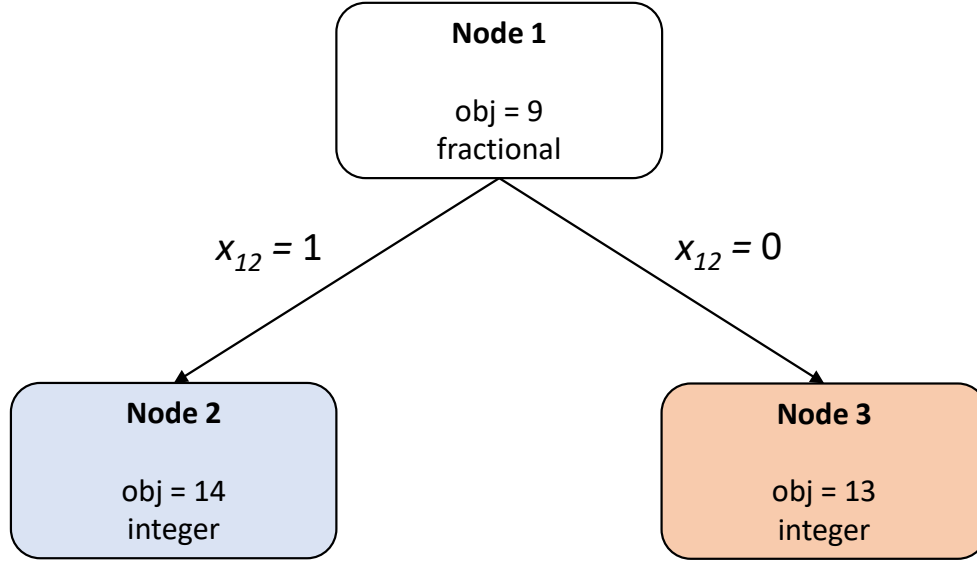


Figure 1: Branch-and-Price Results of Problem 1

The results of the second problem are provided as Figure 2. It gives a more interesting scenario than the first problem. Given that the root node yielded a fractional solution, the first fractional arc (1,2) was chosen to branch. Node 2 resulted in another fractional solution, leading to the creation of Nodes 4 and 5 by respectively setting the decision variable for arc (2,5) to 1 and 0. Subsequently, Node 3 achieved an integer solution. Although Node 4 also produced a fractional solution, it did not undergo further branching as it was dominated by Node 3. Similarly, Node 5 yielded an integer solution and hence did not branch further. Finally, Node 3 emerged as providing the optimal

solution, with a cost of 14 and a path of 1->3->6->10.

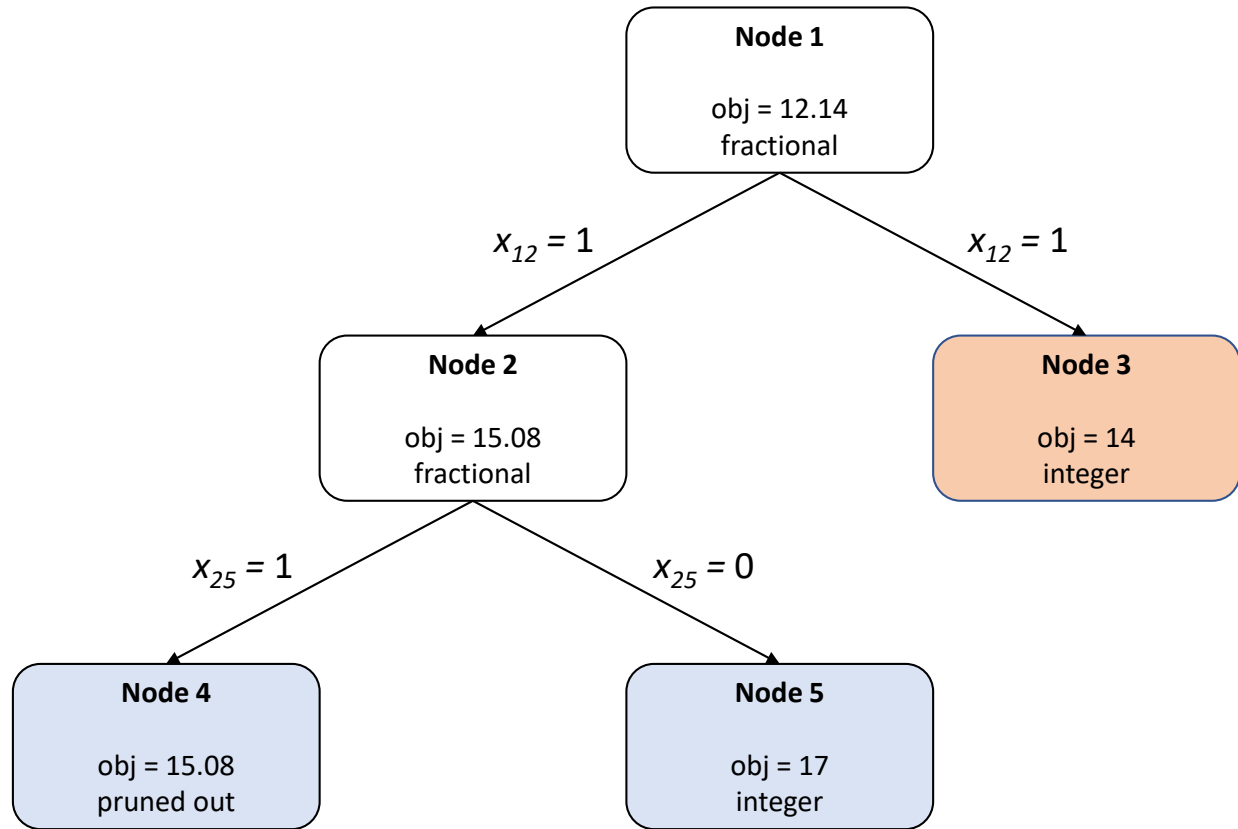


Figure 2: Branch-and-Price Results of Problem 2

4 Conclusions

To summarize,

- Julia functions for solving CSPP were developed in two ways: 1) link/node-based formulation with a general IP solver, 2) Branch-and-Price algorithm
- The solution procedure of Branch-and-Price algorithm was visualized

References

- Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Pearson.
- Desrosiers, J. and Lübbecke, M. E. (2005). *A primer in column generation*. In *Column generation*, pages 132. Springer.
- Handler, G. Y. and Zang, I. (1980). *A dual algorithm for the constrained shortest path problem*. *Networks*, 10(4):293-309.