Capacitated Hub Routing Problem in Hub-and-Feeder Network Design: Modeling and Solution Algorithm

Shahin Gelareh, Rahimeh Neamatian Monemi, Frédéric Semet

Capacitated Hub Routing Problem in Hub-and-Feeder Network Design: Modeling and Solution Algorithm

Shahin Gelareh[1], Rahimeh Neamatian Monemi[2], Frédéric Semet[3]

[1] Université d'Artois, F-62400, Béthune, France
shahin.gelareh@unv-artois.fr
[2] LIMOS UMR 6158-CNRS, Université Blaise-Pascal, BP 10125, F-63173 Aubière Cedex, France
monemi@isima.fr
[3] CRIStAL UMR 9189-CNRS, École Centrale de Lille, Cité Scientifique, F-59650 Villeneuve d'Ascq, France
frederic.semet@ec-lille.fr

Keywords: *Hub Location, Location Routing, Branch-and-Bound, Benders Decomposition.*

1 Introduction

In this paper, we address a new hub location and routing problem. The *Bounded Cardinality Capacitated Hub Routing Problem (BCCHRP)* aims to determine the minimum total times transportation system to transfer goods from origins to destinations through hubs. More precisely, given a set origin-destination (O-D) pairs defined on a set of nodes, we have to partition this set into a subset of hub nodes and a subset of non-hub nodes and to design a network including arcs between hubs and directed routes serving non-hub nodes rooted at hubs. From each hub, one directed route at most can be performed by a vehicle with a limited capacity. The number of hubs used must lie within a minimum and maximum hubs. Transhipment takes place at hub nodes where freight is transferred from the hub-level transporters to the vehicles performing routes. The limited vehicle capacity influences the allocation of non-hub nodes to hub nodes.

The contribution of this paper is twofold. First, we introduce the problem, and propose a polynomial mixed integer linear model. Second, we describe a hybrid exact algorithm combining branch-and-cut methodology and Benders decomposition. The efficiency of the approach is assessed on a various testbed.

Rodriguez-Martìn *et al.* [6] studied a similar problem. In their article, the number of hub nodes is fixed and the capacity restriction is on the maximum number of non-hub nodes served on a route. They proposed a mathematical model and a branch-and-cut approach to solve instances up to 50 nodes. The model and the cuts are based on a previous work devoted to the Plant Cycle Location Problem (PCLP) [5]. Other related papers are the followings. Gelareh *et al.* [4] proposed a mixed integer linear programming formulation and a Lagrangian relaxation for the simultaneous design of network and fleet deployment of a deep-sea liner service provider (*p-String Planning Problem (pSPP)*). Cetiner *et al.* [1] described a multiple allocation hub location and routing problem applied to postal delivery. Again for similar applications, Wasner *et al.* [8] proposed a model where direct connections between non-hub nodes are allowed. de Camargo *et al.* [2] tackled a similar problem where the route lengths are bounded. They proposed a Benders decomposition approach. Last, Nagy *et al.* [7] addressed a hub location routing problem with capacity constraints where pick-up points and delivery points are served on distinct routes.

2 Solution methodology

First, we propose a mixed integer linear model where 2-index variables are associated with the design decisions and 4-index variables are associated with the fractions of flows traversing inter-hub or non-hub arcs for all O-D pairs. Design variables are binary while flow variables are continuous. The model has two sets of constraints: network design constraints and flow routing constraints. To strengthen the linear relaxation of this model, we present several valid inequalities involving the design variables, some of them being similar to those proposed by Rodriguez-Martìn *et al.* [6].

Then, we describe our solution method which is based on a Benders decomposition scheme. The master problem includes the network design constraints while the subproblem includes the flow routing constraints and the linking constraints. The master problem includes also some valid inequalities which are relaxed versions of the capacity constraint imposed on each route. Other cuts are added dynamically. To accelerate the convergence, we use a heuristic based on local searches to generate feasible solutions and add the corresponding cuts to the master problem.

When no violated inequality can be identified, we generate Benders cuts. Due to capacity constraints on routes, the subproblem is still a capacitated multi-commodity flow problem parameterized in the design imposed by the master problem. In some cases, we show that the subproblem can be decomposed into one subproblem per O-D. This leads to disaggregated cuts which can improve the lower bound significantly. In addition, by exploiting symmetry, a mechanism is designed that, under certain conditions, allows extracting a relative interior point of the master problem polytope which is used later on in generating non-dominated cuts from the subproblem.

3 Computational experiments

Based on a testbed generated from the well-known Australian Post (AP) dataset (see [3]), we conducted two types of computational experiments. First, we solved the proposed model with CPLEX 12.6.1 with a time limit of 36000 seconds. Results are reported in Table 1. The first column reports the instance name in the format nN_pP_α where N is the number of nodes, P is the maximum number of hub nodes, and α represents the factor of economies of scale on inter-hub arcs varying in $\{0.7, 0.8, 0.9\}$. The second column indicates CPU times elapsed during the LP solution. The next column reports the LP objective function values followed by a column for the LP statuses. The next column report the number of hub nodes 'Nhubs' in the LP solution. In columns 'TimeIP' and 'IPobj', we report the CPU times for IP solution and the objective function values, respectively. The subsequent column reports the number of nodes processed in the branch-and-bound algorithm before termination criteria is met. The column 'IPStatus' reports the termination criterion that has been met for every instance. The next column reports the optimality gaps when CPLEX terminated.

In Table 1, one observes that the LP bound is rather weak. In the LP relaxation, there is a tendency towards opening maximum possible number of hubs (in the fractional sense). While for $n = 10$ the instances were solved in less than one hour, for $n = 15$, optimality could not be proven within 10 hours.

Our initial computational experiments show that using our solution hybrid solution approach, we

Table 1: Computational experiments with the 2-index model.

instance	TimeRoot	LPobj	LPStatus	Nhubs	TimeIP	IPobj	Nnodes	IPStatus	Gap (%)
n10_p3_0.9	3	2219.32	Optimal	3	1974	3395.63	15905	Optimal	0.00
n10_p3_0.8	3	2160.99	Optimal	3	2177	3315.81	19729	OptimalTol	0.01
n10_p3_0.7	3	2097.88	Optimal	3	3371	3235.99	50926	OptimalTol	0.01
n15_p3_0.9	8	5998.29	Optimal	3	36009	14461.60	2923	AbortTimeLim	56.71
n15_p3_0.8	8	5863.59	Optimal	3	36017	14702.10	4661	AbortTimeLim	56.97
n15_p3_0.7	10	5699.74	Optimal	3	36016	11842.20	4305	AbortTimeLim	48.50
n15_p4_0.9	25	5568.42	Optimal	4	36035	10785.20	5615	AbortTimeLim	43.16
n15_p4_0.8	21	5419.25	Optimal	4	36030	9594.57	5072	AbortTimeLim	37.67
n15_p4_0.7	1-	5240.83	Optimal	4	36025	10425.50	6505	AbortTimeLim	44.79
n15_p5_0.9	20	5203.81	Optimal	5	36021	9122.92	4391	AbortTimeLim	37.44
n15_p5_0.8	17	5039.83	Optimal	5	36023	7381.18	7004	AbortTimeLim	23.58
n15_p5_0.7	20	4852.73	Optimal	5	36027	7885.72	7149	AbortTimeLim	32.88

can solve to optimality up to $n = 20$. The first column in Table 2 indicates the instance name. The second and third columns reports CPU times and the best incumbent value found. The column 'Nnodes' report the number of nodes processed in the course of solution process. The column 'CplexStatus' reports the CPLEX status upon termination. The column 'Gap(%)' reports the termination gaps (the LB is also reported when the method failed). The next two columns represent number of feasibility Benders cuts ('#F. Cuts') and number of optimality Benders cuts ('#O. Cuts').

Table 2: Computational experiments with the Benders decomposition.

instance	Time (sec.)	Obj. Val.	#Nodes	CplexStatus	Gap(%)	#F. Cuts.	#O. Cuts.
n10_p3_0.9	14	3395.63	0	Optimal	0.37	0	696
n10_p3_0.8	14	3315.81	0	Optimal	0.42	0	617
n10_p3_0.7	17	3235.99	0	OptimalTol	0.09	0	785
n15_p3_0.9	398	11244.20	9	OptimalTol	0.19	3	3009
n15_p3_0.8	1395	11255.74	25	OptimalTol	0.27	3	6465
n15_p3_0.7	929	11120.20	47	OptimalTol	0.41	4	5681
n15_p4_0.9	899	9067.25	84	OptimalTol	0.00	1	2690
n15_p4_0.8	883	10170.06	39	OptimalTol	0.41	0	4706
n15_p4_0.7	1073	10683.44	97	OptimalTol	0.37	9	6039
n15_p5_0.9	1615	7387.35	231	Optimal	0.00	0	2364
n15_p5_0.8	591	7143.31	46	Optimal	0.00	0	1989
n15_p5_0.7	626	6899.27	68	Optimal	0.00	0	1581
n20_p3_0.9	28673	24935.81	288	OptimalTol	0.25	10	13250
n20_p3_0.8	—	—	114	failed	22.07%(LB = 19509.37)	—	—
n20_p3_0.7	15307	5548.09	965	OptimalTol	0.46	33	747
n20_p4_0.9	15767	21585.40	184	Optimal	0.00	0	9202
n20_p4_0.8	14432	21284.45	107	OptimalTol	0.48	3	13217
n20_p4_0.7	—	—	199	failed	30.57%(LB = 15169.85)	—	—
n20_p5_0.9	19051	17441.26	301	Optimal	0.00	0	6676
n20_p5_0.8	19572	17624.70	247	OptimalTol	0.24	16	5694
n20_p5_0.7	6470	16933.63	51	Optimal	0.00	0	5984
n20_p6_0.9	98991	17274.12	1525	AbortUser	31.33	0	10308
n20_p6_0.8	46695	16922.59	719	AbortUser	31.01	0	10462
n20_p6_0.7	27531	15738.56	275	OptimalTol	0.41	0	16339

As a rule, the number of nodes processed is very moderate and most instances are solved to optimality. For two instances we have numerical issues that led to failures. The efforts to avoid such numerical instabilities by tuning different parameters and tolerances were not successful. In order to avoid numerical issues we have set the gap tolerance to 0.5%. One observes that the optimality is subject to the user-defined tolerance gap. The number of feasibility Benders cuts corresponds to the iterations where infeasible solution has been encountered. These numbers are also very small. It means that the number of iterations with no (or very minimal) improvement in the lower bound is quite small.

References

[1] Selim Çetiner, Canan Sepil, and Haldun Süral. Hubbing and routing in postal delivery systems. *Annals of Operations Research*, 181(1):109–124, 2010.

[2] Ricardo Saraiva de Camargo, Gilberto de Miranda, and Arne Løkketangen. A new formulation and an exact approach for the many-to-many hub location-routing problem. *Applied Mathematical Modelling*, 37(12):7465–7480, 2013.

[3] A.T. Ernst and M. Krishnamoorthy. Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86(0):141–159, 1999.

[4] Shahin Gelareh, Nelson Maculan, Philippe Mahey, and Rahimeh Neamatian Monemi. Hub-and-spoke network design and fleet deployment for string planning of liner shipping. *Applied Mathematical Modelling*, 37(5):3307–3321, 2013.

[5] Martine Labbé, Inmaculada Rodríguez-Martin, and JJ Salazar-Gonzalez. A branch-and-cut algorithm for the plant-cycle location problem. *Journal of the Operational Research Society*, 55(5):513–520, 2004.

[6] Inmaculada Rodríguez Martín, Juan José Salazar González, and Hande Yaman. A branch-and-cut algorithm for the hub location and routing problem. *Computers & OR*, 50:161–174, 2014.

[7] Gábor Nagy and Said Salhi. The many-to-many location-routing problem. *Top*, 6(2):261–275, 1998.

[8] Michael Wasner and Günther Zäpfel. An integrated multi-depot hub-location vehicle routing model for network planning of parcel service. *International Journal of Production Economics*, 90(3):403–419, 2004.