

Benders Decomposition for the Design of a Hub and Shuttle Public Transit System

Arthur Mahéo, Philip Kilby

The Australian National University, Acton ACT 2601, Australia
DATA61, Tower A, 7 London Circuit, Canberra City ACT 2601, Australia
{Arthur.Maheo, Philip.Kilby}@nicta.com.au,

Pascal Van Hentenryck

The University of Michigan, Industrial and Operations Engineering
1205 Beal Avenue, Ann Arbor, MI 48109.
pvanhent@umich.edu,

The BUSPLUS project aims at improving the off-peak hours public transit service in Canberra, Australia. To address the difficulty of covering a large geographic area, BUSPLUS proposes a hub and shuttle model consisting of a combination of a few high-frequency bus routes between key hubs and a large number of shuttles that bring passengers from their origin to the closest hub and take them from their last bus stop to their destination.

This paper focuses on the design of bus network and proposes an efficient solving method to this multimodal network design problem based on the Benders decomposition method. Starting from a MIP formulation of the problem, the paper presents a Benders decomposition approach using dedicated solution techniques for solving independent sub-problems, Pareto optimal cuts, cut bundling, and core point update. Computational results on real-world data from Canberra's public transit system justify the design choices and show that the approach outperforms the MIP formulation by two orders of magnitude. Moreover, the results show that the hub and shuttle model may decrease transit time by a factor of 2, while staying within the costs of the existing transit system.

Key words: combinatorial optimisation, multimodal transportation, Benders decomposition, Pareto optimal cuts, cut bundling

1. Introduction

Canberra is a planned city designed by American architect Walter Griffin in 1913. It features a large number of semi-autonomous towns separated by greenbelts. As a result, Canberra covers a wide geographic area, which makes public transportation particularly challenging. Bus routes are long and hence bus frequencies, and patronage, are low, especially during off-peak periods. To address these imitations, the BUSPLUS project designed, optimized, and simulated a Hub and Shuttle Public Transit System. The Hub and Shuttle model consists of a combination of a few high-frequency bus routes between key hubs and a large number of shuttles (or multi-hire taxis) that bring passengers from their origin to the closest hub and take them from their last bus stop to their destination.

The main advantage of BUSPLUS is its ability to deliver the same service regardless of the origin and destination of a passenger. BUSPLUS uses bus stops instead of a *doorstep* shuttle service to reduce the delay of waiting for the customer to exit her house, e.g., putting on shoes, coat, ... In addition, bus stops provide an already established network covering the city, within walking distance of homes even in its remotest areas. From a traveller standpoint, this transit model is highly convenient: Travellers book their travel online (e.g., on their phones), are picked up at their traditional bus stop, and dropped at their destination for the same ticket price as before. The anticipated benefits come from the level of service: The hope is to reduce travel time significantly for the same overall system cost.

Designing such a Hub and Shuttle Public Transit System (HSPTS) however creates a series of interesting challenges, including

1. How to connect a set of potential hubs to minimize costs and maximize convenience?
2. How to allocate, during operations, requests to a trip, consisting of a shuttle leg, a number of bus legs, and a final shuttle leg?

This paper focuses on the first problem and primarily on off-peak hours – evenings, week-ends –, which are the most challenging from a cost and service standpoint. Designing a HSPTS differs from the traditional bus-network design problem which, given a set of known or implied origin/destination demands, consists in building a network of routes that visit all the bus stops and serve these demands. In contrast, a solution to the HSPTS does not need to visit all the bus stops nor even all the potential hubs. *The HSPTS goal is to decide which hubs to link via bus routes to take advantage of economies of scale, while relying on shuttles for the remaining "last mile" elements of the service.* As a result, in a HSPTS system, the design of the bus network is intertwined with the routing of passengers: The objective is to balance the cost of running buses for a whole day, which has a high up-front cost, with the cost of routing passengers with shuttles only, which has a low fixed cost but incurs a costly per trip expense.

The paper models the HSPTS design problem as an Hub-Arc Location Problem (HALP) (Campbell et al. 2005a,b) and uses Benders decomposition (Benders 1962) to solve the resulting MIP formulation. The Benders decomposition makes use of the Pareto-optimal cuts proposed by Magnanti and Wong (1981), core point updates, and cut bundling using the problem structure. The experimental results, based on real data collected on the Canberra public transit system, show the benefits of the HSPTS proposed by the BUSPLUS project, as well as the effectiveness of Benders decomposition and its various enhancements. Observe also that, although the main motivation of the BUSPLUS project concerns the off-peak setting, the experimental results are based on actual data covering a whole day of operations to validate the scalability of the model.

The rest of the paper is organized as follows. Section 2 describes the HSPTS model considered in this paper, its simplifying assumptions, and the data sets. Section 3 reviews prior work on related problems. Section 4 reviews the MIP formulation and a number of problem-specific pre-processing techniques. Section 5 reviews Benders decomposition and some of its important extensions. Section 6 presents the Benders decomposition approach. Finally, Section 7 presents the experimental results for the case study, including the benefits of the Benders decomposition and the impact of the BUSPLUS project on the public transit system in Canberra.

In the following, we use shuttles and taxis interchangeably, since the shuttles in our case study are multi-hire taxis, which are available in large numbers in Canberra.

2. Modeling the HSPTS Problem for the BusPlus Project

The HSPTS is a variant of the hub-arc location problem in which customers have access to a multimodal transportation where buses run on a fixed network and taxis can transport customers on any origin-destination pair. Bus routes can be opened for a fixed cost which represents the cost of operating high-frequency buses along the arc. The goal is to minimize the cost of system, i.e., the fixed cost of operating the bus lines and the variable cost for each taxi trip, together with maximizing the convenience for the travellers. We use the trip duration as a proxy for traveller convenience in the model.

To make the design of the HSPTS manageable from a computational standpoint, a number of simplifications are introduced. First, trips are modeled as single commodities and the number of passengers is a factor on the cost of the trip. Second, the routing and scheduling aspects are ignored. Taxis are considered to be available at any location within a short time and always travel directly from pickup to setdown. This assumption is reasonable whenever the HSPTS system is able to call on a sufficiently large taxi fleet, which is certainly the case in Canberra. Buses are considered to be available for connections with a nominal waiting time. The model does not account for capacities, which is again a reasonable assumption in the off-peak times. Obviously, the BUSPLUS project has also developed online optimization algorithms for taxi dispatching but this is outside the scope of this paper.

Our dataset represents a week worth of trips in Canberra using the current public transit network. Each trip has an origin and a destination and a number of passengers. Time and distance matrices between each pair of nodes gives the on-road distance and average travel time between each pair of nodes: They both respect the triangle inequality. Finally, Figure 1 presents the two sets of potential hubs to be used for building the bus network: H_{10} with ten hubs and H_{20} with twenty hubs. The hubs are shown by blue dots in the figure. The hubs were chosen using a *p-median* model (Hakimi 1964).

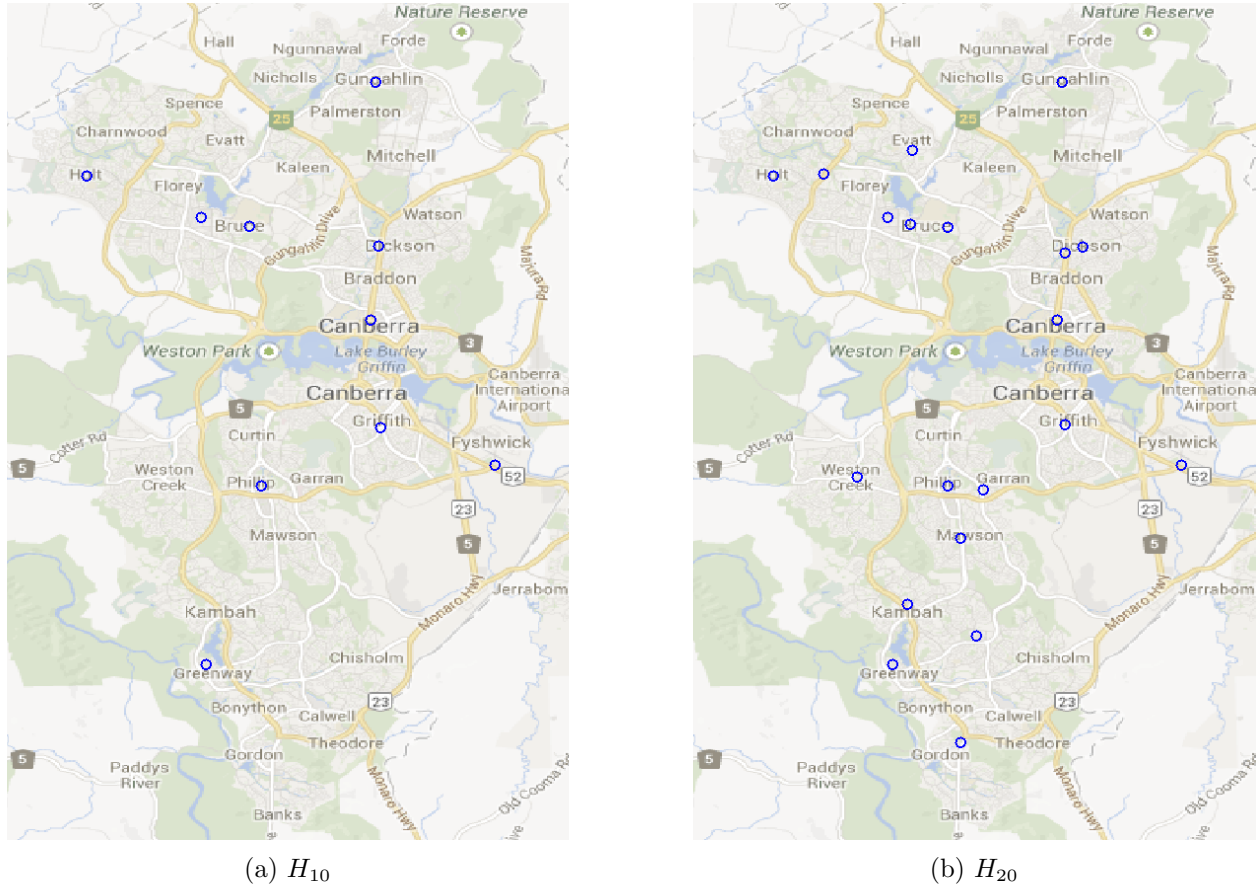


Figure 1 Two Potential Configurations for BUSPLUS with 10 and 20 Hubs Respectively.

3. Review of Prior Work

The problem of linking a set of hubs using arcs is called the Hub-Arc Location Problem or *hub-and-spoke* network design problem: It was introduced by Campbell et al. (2005a,b) and is defined as the problem of locating a given number of hub arcs in such a way that the total flow cost is minimized. It is mostly used in transshipment contexts where economies of scale can be expected by grouping flows. The Hub-Arc Location Problem (HALP) can be seen as a two-level decision problem deciding which arcs to open first and then how to route the flow at minimum cost. As such, its structure appears ideally suited for Benders decomposition (Benders 1962). In recent years, a significant body of research has been dedicated to solving variants of the Hub Location Problem and the HALP using Benders Decomposition. These studies typically impose restrictions on the network topology. They include, for instance, the Tree Hub Location Problem (de Sá et al. 2013), where hubs are connected in a non-directed tree, the Hub Line Location Problem (de Sá et al. 2015a), where the decision is to locate q lines connecting hubs, and star topologies (Labbé and Yaman 2008, Yaman 2008) where all hubs are connected to a central hub and nodes are connected to hubs. de Sá et al. (2015b) introduced the Hub Line Location Problem which consists of locating p hubs and connecting them by means of a path, i.e., a single line. The arcs linking the hubs are a

way to speed up travel time but commodities can also be routed outside the network. There may also be a cost to access the network. In order to solve this complex problem, de Sá et al. (2015b) used a Benders decomposition that include multicuts, a dedicated algorithm for the sub-problem, and the use of a branch-and-cut framework.

Marín and Jaramillo (2009) solve a rapid urban transit network design problem, where the goal is to locate train stations in an uncapacitated network with line location and demand routing constraints. Moreover, customers have access to a *private* network in competition with the public network of trains. They improve the convergence speed of the Benders algorithm by using sub-problem disaggregation, a procedure to remove inactive optimality cuts, and a dedicated sub-problem algorithm. The goal in this work is to maximize the demand served by the transit system, while our goal is to optimize a combination of cost and convenience.

Finally, it is useful to mention that nodes can be assigned to hubs in two main ways: single or multiple allocations. With multiple allocations, the commodities from a node can be sent to multiple hubs. While the single allocation is the most commonly studied, de Camargo et al. (2008) apply Benders decomposition to the uncapacitated multiple-allocation HALP. The main issue is the non-linearity of their model. As a result, they use a Generalised Benders Decomposition with a MIP as the master problem and a non-linear convex transportation problem as the subproblem. Note that de Sá et al. (2015b) also allow for multiple allocations.

4. A MIP Model

This section presents a MIP model for the HSPTS problem. It starts by presenting the main formulation and then proposes a number of domain-specific pre-processing steps.

4.1. The Basic Formulation

4.1.1. Inputs The inputs to the HSPTS are as follows:

1. a complete graph G with a set N of nodes;
2. a set T of trips to serve, where each trip $r \in T$ is specified by a tuple (o, d, p) with an origin $o^r \in N$, a destination $d^r \in N$, and a number of passengers $p^r \in \mathbb{N}$;
3. a subset $H \subseteq N$ of nodes that can be used as bus hubs.

In the following, we use $or(r)$ and $de(r)$ to denote the origin and destination of a trip r . The distance and the travel time from node i to j ($i, j \in N$) are given by d_{ij} and t_{ij} respectively. Since the HSPTS problem aims at optimizing cost and convenience jointly, we use α as a *conversion factor* to translate travel times into financial costs. Travel distance is used as a cost measure, while travel time is a proxy for convenience. The value α shifts the solution towards cost or convenience by modifying travel times by a factor α and travel costs by $(1 - \alpha)$: The lower α is, the more cost-driven the solution is. The objective is formulated in terms of the following constants:

- c : the cost of using a taxi per kilometer;
- b : the cost of using a bus per kilometer;
- n : the number of buses per day which depends on the headway;
- S : the average waiting for a bus.

The different characteristics of the two transportation modes are captured in their associated cost functions: The cost (τ) of a taxi is a combination of a cost per kilometer and a cost per minute, while the cost (γ) of bus is only a function of time. However, buses run for the whole day, which is modelled by a large initial set-up cost (β). These costs are defined as follows:

- $\tau_{ij} = (1 - \alpha)c \cdot d_{ij} + \alpha \cdot t_{ij}$ (cost and convenience of using a taxi from i to j);
- $\gamma_{hl} = \alpha(t_{hl} + S)$ (convenience of using a bus from h to l);
- $\beta_{hl} = (1 - \alpha)b \cdot n \cdot d_{hl}$ (cost of opening a bus leg from h to l).

Since all the passengers in a trip r are travelling at the same time, the taxi and bus costs of r are obtained by multiplying the arc costs by the number of passengers, i.e.,

- $\tau_{ij}^r = p^r \cdot \tau_{ij}$;
- $\gamma_{hl}^r = p^r \cdot \gamma_{ij}$.

4.1.2. Decision Variables The decision variables for the HSPTS problem, which are all binary, are as follows: Binary variable x_{ij}^r denotes whether trip r uses a taxi to travel arc (i, j) , variable y_{hl}^r denotes whether trip r uses a bus on arc (h, l) ($h, l \in H$), and variable z_{hl} indicates whether arc (h, l) is opened for buses to use. By convention, this paper uses (i, j) for edges travelled by taxis and (h, l) for edges used by buses.

4.1.3. Network Topology The MIP model only enforces a weak form of connectivity that requires that the sum of all incoming bus legs must be equal to the sum of all outgoing bus legs at every hub. In other words, each leg entering a hub has a corresponding leg which has to leave the hub. While this formulation technically leaves open the possibility of producing a network of disconnected components, the demand patterns observed in practice lead to networks of good quality. With low demand, the topology is often a simple circuit around the center. Higher demand patterns often produces one or more connected flower topologies with sub-circuits extending from the center to the suburbs. Figure 10 in Section 7 illustrates these topologies.

4.1.4. The MIP Model We are now in a position to present the MIP model for the HSPTS of BUSPLUS. This model has similarities with the HAL4 from Campbell et al. (2005a) but it relaxes a number of critical assumptions for our case study, i.e.,

1. An optimal solution may contain a path from the origin to the destination that does not contain a hub.
2. There is no constraint on the number of arcs to open;

$$\begin{aligned}
 \min \quad & \sum_{r \in T} \sum_{i, j \in N} \tau_{ij}^r x_{ij}^r + \sum_{r \in T} \sum_{h, l \in H} \gamma_{hl}^r y_{hl}^r + \sum_{h, l \in H} \beta_{hl} z_{hl} & \text{(MIP)} \\
 \text{s.t.} \quad & \sum_{l \in H} z_{hl} = \sum_{l \in H} z_{lh} & \forall h \in H & \text{(1a)} \\
 & y_{hl}^r \leq z_{hl} & \forall r \in T, \forall h, l \in H & \text{(1b)} \\
 & \sum_{j \in N} (x_{ij}^r - x_{ji}^r) + \sum_{\substack{h \in H \\ \text{if } i \in H}} (y_{ih}^r - y_{hi}^r) = \begin{cases} 1 & \text{if } i = or(r) \\ -1 & \text{if } i = de(r) \\ 0 & \text{otherwise} \end{cases} & \forall r \in T, \forall i \in N & \text{(1c)} \\
 & x_{ij}^r, y_{hl}^r, z_{hl} \in \mathbb{B}
 \end{aligned}$$

Figure 2 The MIP Model for the BUSPLUS HSPTS.

3. The model allows for bridge arcs between two hubs.

The MIP model is presented in Figure 2. The objective function is the sum of the travelling cost of the trips using their selected arcs and the cost of opening the bus legs. Constraints (1a) enforce the weak connectivity constraints on the bus legs, Constraints (1b) ensure that travellers only use opened bus legs, and Constraints (1c) enforce flow conservation, i.e., travellers start at their origin and reach their destination without skipping network edges. Note that, once the z variables are fixed, the problem becomes totally unimodular and the integrality constraints on the x and y variables can be relaxed.

4.2. Problem-Specific Preprocessing

We now describe two filtering techniques that decrease the number of decision variables significantly: Trip filtering and link filtering.

4.2.1. Trip Filtering It may be the case that, in all possible configurations of the HSPTS problem, the optimal routing of a trip r is a direct taxi ride. These trips can be filtered from the HSPTS problem, since they do not impact its optimal solutions. Such trips can be identified by a simple filtering algorithm that

1. considers that all bus legs are open;
2. computes a least-cost path from the source to the destination of each trip;
3. removes the trip if the least-cost path is a direct taxi ride.

It is interesting to observe that, in the filtering procedure, a least-cost trip can be a direct taxi ride or one of the four patterns depicted in Figure 3, since it is assumed that all bus legs are open. In particular, a least-cost trip is either (1) a single taxi ride or (2) a journey with exactly one bus leg and possibly a taxi ride from the origin to the hub and/or from the hub to the destination.

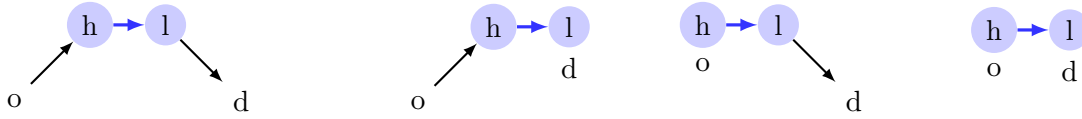


Figure 3 Possible Trip Patterns (Not Showing Direct Taxi Trips).

	Total	H_{10}		H_{20}	
		T-Filtered	Reduction (%)	T-Filtered	Reduction (%)
Monday	21282	14324	32.69	17294	18.74
Tuesday	21029	14184	32.55	17084	18.76
Wednesday	21418	14451	32.53	17401	18.76
Thursday	21487	14486	32.58	17499	18.56
Friday	19809	13398	32.36	16013	19.16

Table 1 Effectiveness of Trip Filtering. Column *Total* gives the initial number of trips. Columns *T-Filtered* report the number of trips after trip filtering. Columns *Reduction* give the trip reduction in percentage.

Table 1 reports experimental results on the effectiveness of trip filtering. Column *Total* reports the initial number of trips, while the *T-Filtered* columns give the trips remaining after filtering for both hub configurations. Column *Reduction* report the trip reduction in percentage for both configurations. For the 10 hub configuration, the filtering reduces the number of trips by more than 30%. The reduction is more than 18% in the configuration with 20 hubs. This lower number was of course expected for 20 hubs, since additional hubs make more complex hub patterns more attractive from a cost standpoint.

4.2.2. Link Filtering The current formulation considers every possible taxi link, thus using a complete graph. Because of the triangle inequality, there is no need to connect all the nodes using taxis: The only taxi links to consider for a trip $r = (o, d, p)$ are

1. from the origin to a hub;
2. from a hub to the destination;
3. and from the origin to the destination.

As a result, the formulation needs only the following taxi variables:

$$x_{oh}^r \geq 0 \quad \forall r \in T, \forall h \in H;$$

$$x_{hd}^r \geq 0 \quad \forall r \in T, \forall h \in H;$$

$$x_{od}^r \geq 0 \quad \forall r \in T.$$

Observe that τ_{od} , the cost of a one-person direct taxi trip, is an upper bound to the cost of a trip. This upper bound can be used to filter links from the origin o to a hub h and from a hub l to the destination d by generalizing the trip filtering presented earlier. Indeed, if the least-cost path going through (o, h) is not cheaper than τ_{od} , variable x_{oh}^r can be removed. Consider, for instance,

	H_{10}			H_{20}		
	T-Filtered	L-Filtered	Reduction (%)	T-Filtered	L-Filtered	Reduction (%)
Monday	292958	131580	55.09%	698910	294726	57.83%
Tuesday	290100	129998	55.19%	690278	291736	57.74%
Wednesday	296000	133533	54.89%	706873	299117	57.68%
Thursday	295555	132500	55.17%	703205	296852	57.79%
Friday	273226	124953	54.27%	646095	279811	56.69%

Table 2 Effectiveness of Link Filtering: The T-Filtered and L-Filtered columns give the number of taxi variables left after applying the T-filtering and L-filtering procedures. The reduction columns give the percentage of taxi variables pruned by L-filtering from the T-filtered instances.

the case where the trips have three components: 1) a taxi trip from the origin to a hub; 2) a bus trip between two hubs; and 3) a taxi trip from the last hub to the destination. If the condition

$$\forall l : \tau_{oh} + \gamma_{hl} + \tau_{hl} > \tau_{od}$$

holds, then the taxi trip (o, h) will never be used and the variable x_{oh}^r can be removed. The symmetric condition

$$\forall h : \tau_{oh} + \gamma_{hl} + \tau_{hl} > \tau_{od}$$

allows to remove variable x_{ld}^r . Similar reasonings can be applied to the other patterns depicted in Figure 3

We call this procedure link filtering (L-filtering for short). Once the T-filtering and L-filtering procedures have been applied, the remaining taxi arcs are called *useful* and D^r denotes the set of useful taxi arcs for a trip r . Table 2 presents experimental results on link filtering. They indicate that the link filtering procedure is particularly effective, removing more than 50% of the taxi arcs available after trip filtering. As the table shows, this holds for the two configurations with 10 and 20 hubs respectively.

5. A Brief Review of Benders Decomposition

This section provides a brief overview of Benders decomposition for solving MIPs (Benders 1962). The overview uses some simplifying assumptions that hold in the HSTPS problem. Assume, for simplicity, that the MIP model is specified as follows:

$$\min \quad c^T x + f^T y \quad (2a)$$

$$s.t. \quad Ax + By \geq b \quad (2b)$$

$$x \geq 0, y \in \mathbb{Y} \quad (2c)$$

where the x variables are real-valued and the y variables are discrete and belong to the feasible set \mathbb{Y} . Benders decomposition splits the MIP model into two parts: a master problem that contains

the y variables and a sub-problem that contains the x variables and a candidate solution for the y variables. The subproblem can be specified as

$$q(y) = \min \quad c^T x \quad (3a)$$

$$s.t. \quad Ax \geq b - By \quad (3b)$$

$$x \geq 0 \quad (3c)$$

and the master problem becomes

$$\min \quad f^T y + q(y) \quad (4a)$$

$$y \in \mathbb{Y} \quad (4b)$$

For simplicity, we assume that the subproblem (3) is always feasible and bounded, which will be the case for the HSTPS problem. Let α be the dual variables associated with (3b). The dual of (3) is given by

$$\max \quad \alpha^T (b - By) \quad (5a)$$

$$s.t. \quad A^T \alpha \leq c \quad (5b)$$

$$\alpha \geq 0 \quad (5c)$$

Let π_j ($j \in [1, J]$) be the extreme points of the dual feasibility region, which does not depend on the candidate values for the y variables. The subproblem (3) can be reformulated as

$$\min \quad q \quad (6a)$$

$$s.t. \quad \pi_j^T (b - By) \leq q \quad \forall j \in [1, J] \quad (6b)$$

$$q \in \mathbb{R} \quad (6c)$$

and the master problem as

$$\min \quad f^T y + q \quad (7a)$$

$$s.t. \quad \pi_j^T (b - By) \leq q \quad \forall j \in [1, J] \quad (7b)$$

$$q \in \mathbb{R}, y \in \mathbb{Y} \quad (7c)$$

where the constraints (7b) are called Benders (optimality) *cuts*.

Since there are potentially an exponential number of extreme points, Benders decomposition starts by solving a restricted master problem with no cuts and introduces the Benders cuts lazily. At each iteration, Benders decomposition uses the restricted master to produce candidate solution (\bar{y}, \bar{q}) and then solves the subproblem $q(\bar{y})$. If $q(\bar{y}) = \bar{q}$, the candidate solution is optimal. Otherwise, a Benders cut of the form (7b) is added to the restricted master problem and the process is repeated.

5.1. Separable Benders Subproblem and Cut Bundling

Consider now the case where the subproblem (3) is separable, i.e., it can be rewritten as

$$\min \quad c_1^T x_1 + c_2^T x_2 + \cdots + c_n^T x_n \quad (8a)$$

$$s.t. \quad A_1 x_1 \geq b_1 - B_1 y_1 \quad (8b)$$

$$A_2 x_2 \geq b_2 - B_2 y_2 \quad (8c)$$

⋮

$$A_n x_n \geq b_n - B_n y_n \quad (8d)$$

$$x_i \geq 0 \quad \forall i \in [1, n]$$

Solving the subproblem now consists of optimizing independent components, i.e.,

$$\begin{aligned} \min \quad & c_1^T x_1 & \min \quad & c_2^T x_2 \\ (8) \Leftrightarrow s.t. \quad & A_1 x_1 \geq b_1 - B_1 y_1 & + s.t. \quad & A_2 x_2 \geq b_2 - B_2 y_2 & + \dots \\ & x_1 \geq 0 & & & & x_2 \geq 0 \end{aligned}$$

A separable subproblem gives some new possibilities for cut generation. Indeed, Geoffrion and Graves (1974) observe that, in this case, there exist better strategies than aggregating the results of the sub-optimizations to produce a single cut (see also the concept of *multi-cut* proposed by Birge and Louveaux (1988)). We will investigate various *cut bundling* strategies for the HSTPS problem in the next section.

5.2. Pareto Optimal Cuts

When the subproblem is a network flow optimization, which exhibits high degeneracy (Ahuja et al. 1988), Benders decomposition may experience slow convergence and it becomes important to generate stronger, *Pareto optimal*, cuts as proposed by Magnanti and Wong (1981). Given two solutions π_a and π_b to (3), π_a dominates π_b if

$$\pi_a^T (b - By) \geq \pi_b^T (b - By),$$

and at least one of these inequalities is strict. A solution (and its associated cut) that dominates all other solutions is said to be Pareto optimal. Magnanti and Wong (1981) showed how to generate a Pareto optimal cut by solving an optimization problem, using the notion of a *core point* of a set S , i.e., a point in the relative interior of the convex hull of S denoted by $ri(S)$. A Pareto optimal cut can be obtained by solving the following optimization problem:

$$\max \quad \alpha^T (b - By^0) \quad (9a)$$

$$s.t. \quad A^T \alpha \leq c \quad (9b)$$

$$\alpha^T (b - B\bar{y}) = q(\bar{\alpha}) \quad (9c)$$

$$\alpha \geq 0 \quad (9d)$$

where \bar{y} is the candidate solution from the restricted master problem, $\bar{\alpha}$ is an optimal solution to (5), and $y^0 \in ri(\mathbb{Y})$ is a core point of \mathbb{Y} .

Papadakos (2009) showed that the choice of a core point can dramatically improve the convergence rate of Benders decomposition. Moreover, the author proposed a method for computing a sequence of core points by using a linear combination of the current core point and the solution obtained at each iteration of Benders decomposition. Experimental results have shown that using weights of 1/2 gives excellent results.

6. Benders Decomposition for the HSTPS Problem

This section presents how to apply Benders decomposition to the HSTPS problem. The key idea is to assign the z variables in the MIP model to the restricted master problem, leaving the x and y variables for the subproblem. As mentioned earlier, when the z variables are fixed to \bar{z} , the subproblem is a min-cost flow, which can be solved efficiently. The subproblem can be specified as

$$\min \quad \sum_{r \in T} \left(\sum_{i,j \in N} \tau_{ij}^r x_{ij}^r + \sum_{h,l \in H} \gamma_{hl}^r y_{hl}^r \right) \quad (10a)$$

$$s.t. \quad \sum_{j \in H} (x_{ij}^r - x_{ji}^r) + \sum_{\substack{h \in H \\ \text{if } i \in H}} (y_{ih}^r - y_{hi}^r) = \begin{cases} 1 & \text{if } i = or(r) \\ -1 & \text{if } i = de(r) \\ 0 & \text{otherwise} \end{cases} \quad \forall r \in T, \forall i \in N \quad (10b)$$

$$y_{hl}^r \leq \bar{z}_{hl} \quad \forall r \in T, h, l \in H \quad (10c)$$

$$0 \leq x_{ij}^r, y_{hl}^r \leq 1$$

The dual of the problem can be specified in terms of the dual variables u_i^r associated with constraints (10b) variable and the dual variables v_{hl}^r associated with constraints (10c).

$$\max \quad \sum_{r \in T} (u_o^r - u_d^r) - \sum_{h,l \in H} z_{hl} \left(\sum_{r \in T} v_{hl}^r \right) \quad (11a)$$

$$s.t. \quad u_i^r - u_j^r \leq \tau_{ij}^r \quad \forall r \in T, i, j \in D^r \quad (11b)$$

$$u_h^r - u_r^r - v_{hl}^r \leq \gamma_{hl}^r \quad \forall r \in T, h, l \in H \quad (11c)$$

$$u_i^r \geq 0, v_{hl}^r \geq 0$$

Note that Model (10) is separable: Each trip r gives rise to an independent subproblem.

6.1. Cut Aggregation

We now discuss how to generate the cuts for the restricted master problem. Since the subproblem is separable, their individual cuts can be aggregated in many ways and we explored a variety of cut bundling strategies that exploit the structure of the HSTPS problem. For instance, a bundling

strategy may aggregate the cuts for all the trips with the same origin. It is also important to mention that each bundled cut must create its own variable and that each original variable should appear in exactly one cut. The following cut bundling strategies were investigated for the HSTPS problem (in the cut definitions, \bar{u}_i^r and \bar{v}_{hl}^r denote the values of the dual variables in the subproblem).

One The traditional Benders cut from (10).

$$\sum_{r \in T} (\bar{u}_o^r - \bar{u}_d^r) - \sum_{h, l \in H} z_{hl} \left(\sum_{r \in T} \bar{v}_{hl}^r \right) \leq q \quad (12)$$

Multi This strategy does not aggregate the cuts.

$$\bar{u}_o^r - \bar{u}_d^r - \sum_{h, l \in H} z_{hl} \left(\sum_{r \in T} \bar{v}_{hl}^r \right) \leq q^r \quad \forall r \in T \quad (13a)$$

$$\sum_{r \in T} q_r \leq q \quad (13b)$$

Hubs This strategy aggregates the subproblems by the closest hub to a trip origin or destination. Let T_h be the set of trips having h as their closest hub.

$$\sum_{r \in T_h} (\bar{u}_o^r - \bar{u}_j^r) - \sum_{l \in H} z_{hl} \left(\sum_{r \in T_h} \bar{v}_{hl}^r \right) \leq q_h \quad \forall h \in H \quad (14a)$$

$$\sum_{h \in T_h} q_h \leq q \quad (14b)$$

Origin This strategy aggregates the subproblems by trip origins. Let T_o be trips having origin o and O be the set of all possible origins.

$$\sum_{r \in T_o} (\bar{u}_o^r - \bar{u}_j^r) - \sum_{h, l \in H} z_{hl} \left(\sum_{r \in T_o} \bar{v}_{hl}^r \right) \leq q_o \quad \forall o \in O \quad (15a)$$

$$\sum_{o \in O} q_o \leq q \quad (15b)$$

Legs This strategy aims at grouping the trips by the first bus leg they use. Since this bus leg is not known a priori, the leg strategy aggregates the subproblems by the bus leg used in trip filtering. Recall that all trips are associated with a single bus leg in trip filtering since all the bus legs are open. Let T_{hl} be the set of trips using bus leg (h, l) in the trip filtering.

$$\sum_{r \in T_{hl}} (\bar{u}_o^r - \bar{u}_j^r) - z_{hl} \sum_{r \in T_{hl}} \bar{v}_{hl}^r \leq q_{hl} \quad \forall h, l \in H \quad (16a)$$

$$\sum_{h, l \in H} q_{hl} \leq q \quad (16b)$$

6.2. Restricted Master Problem

We are now in a position to present the restricted master problem, whose objective minimizes the cost of opening the bus legs subject to the circular constraint (1a) and one of the cut sets defined above for each iteration:

$$\begin{aligned} \min \quad & \sum_{h,l \in H} \beta_{hl} z_{hl} + q \\ \text{s.t.} \quad & (1a), \text{ one of } (12) - (16) \\ & z \in \mathbb{B}, q \in \mathbb{R} \end{aligned}$$

6.3. Pareto-Optimal Cuts

Our Benders implementation for the HSTPS problem uses Pareto-optimal cuts, which requires to solve two linear programs: The original sub-problem (10) and then the Pareto sub-problem.

Observe first that each of the independent subproblems is a min-cost flow with edges of infinite capacity and is equivalent to a shortest path problem. Hence, the original subproblem for a given trip can be solved by computing, for each trip, a shortest path between the origin and the destination using the edges defined by the union of D^r , the useful taxi arcs, and \bar{Z} , the set of opened bus legs in the current iteration.

To define the Pareto subproblem, it is necessary to find a core point that satisfies the circular constraint (1a). It can be chosen easily by assigning the same value to all z variables: i.e.,

$$z_{hl}^0 = \zeta \quad \forall h, l \in H, \zeta \in]0, 1[\quad (18)$$

The Pareto subproblem for a trip r then becomes

$$\max \quad u_o^r - u_d^r - \sum_{h,l \in H} z_{hl}^0 v_{hl} \quad (\text{Pareto})$$

$$\text{s.t.} \quad u_i^r - u_j^r \leq \tau_{ij} \quad \forall i, j \in D \quad (19a)$$

$$u_h^r - u_l^r - v_{hl}^r \leq \gamma_{hl} \quad \forall h, l \in H \quad (19b)$$

$$u_o^r - u_d^r - \sum_{h,l \in H} \bar{z}_{hl} v_{hl} = \sigma \quad (19c)$$

$$u_i^r \geq 0, v_{hl}^r \geq 0$$

where σ is the optimal objective value to the original subproblem for trip r .

Our implementation also updates the core point, which can be seen as an intensification procedure: Seldom used bus legs decay towards low values while bus legs present in every solution are assigned a high coefficient in Pareto solutions. The update rule is defined as follows:

$$z_{hl}^{0(k+1)} = \frac{z_{hl}^{0(k)} + \bar{z}_{hl}^{(k)}}{2} \quad \forall h, l \in H. \quad (20)$$

c (\$/km)	b (\$/km)	S (s)	n	α	Hubs
1.96	4.5	30	32	10^{-3}	H_{10}

Table 3 The Default Values of the Parameters.

7. Experimental Results

This section presents experimental results on the HSTPS problem. It starts by specifying the experimental setting. It continues by justifying the implementation choices, including the Benders scheme, the cut bundling strategy, and the core point updating rule. It also compares the final algorithm with a standard MIP approach. Finally, the section evaluates the impact of the algorithm on the real case-study that motivated this work: The restructuring of the Canberra public transit system.

7.1. Experimental Setting

The results in Section 4.2 already indicated that the difference between the weekdays is minimal. As a result, the experimental results use a single day and two sets of instances: A set of small instances with a number of trips ranging from 100 to over 2,000 trips; and a set of large instances ranging from 1,000 to over 20,000 trips. The small instances are primarily used to eliminate certain algorithmic options. The large instances are useful to demonstrate scalability of the various advanced features. Unless specified otherwise, the algorithms are used with the parameter settings presented in Table 3. Note that the time penalty emulates the waiting time associated with any bus trip, i.e., it includes at least ten minutes of waiting time – five minutes at the first and last stops.

The LP/MIP solver used in the experiments in Gurobi v6.0 (Gurobi Optimization 2015) with default parameters using the Dual Simplex and without pre-processing. Gurobi was used to solve the MIP formulation, the Benders master problem, and the sub-problems. The algorithms are implemented in C++ and run as single threaded programs, forcing Gurobi to use a single thread as well. The experiments were run on a cluster with AMD Opteron 4184 CPUs and 64GB of RAM.

7.2. Justification of the Benders Approach

7.2.1. Benders’s Schemes Figure 4 reports results of the following implementations of Benders decomposition:

CR : A single standard subproblem;

CP : A single Pareto subproblem;

SR : Multiple independent standard subproblems;

SP : Multiple independent Pareto subproblems.

The results are for the small set of instances and use the standard parameters. The algorithms have a limit of 100 iterations: If the Benders decomposition needs more than 100 iterations, it is most likely exhibiting significant degeneracy.

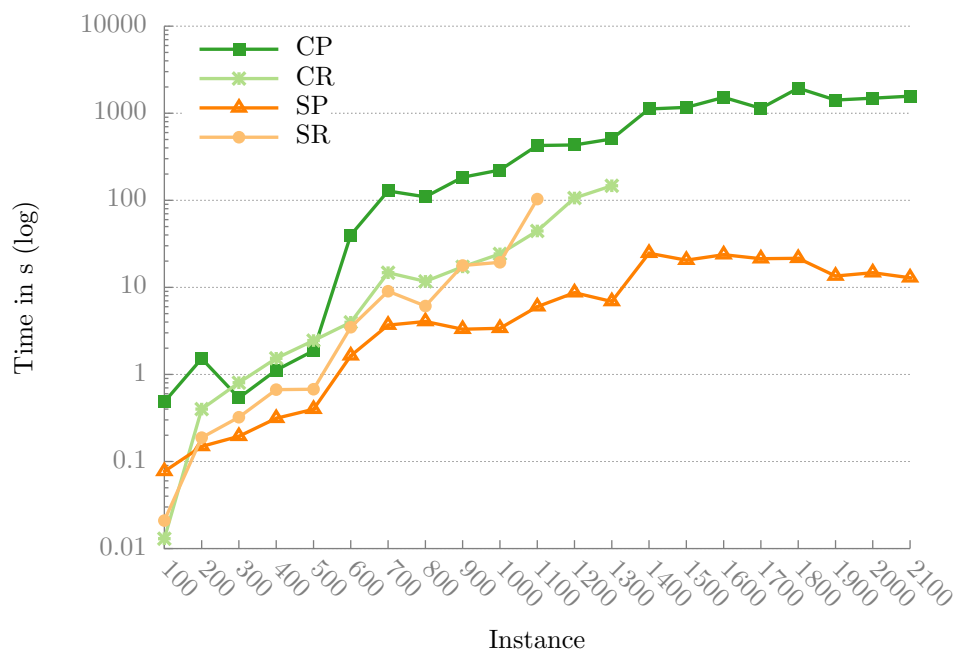


Figure 4 Comparison of the Benders Decomposition Schemes.

The results in Figure 4 clearly show the value of Pareto-optimal cuts as the standard approach can only solve the lower half of the instances before requiring more than a hundred iterations. They also show that a single Pareto subproblem creates a hard MIP problem and execution times become a significant issue. Solving independent Pareto subproblems addresses both issues.

7.2.2. Cut Aggregation Schemes Figure 5 presents the results comparing the cut aggregation schemes on the small instances, while Figure 6 depicts the behavior of three bundling strategies, one cut, hub aggregation, and leg aggregation, on the large instances. They both present the average runtime per instance; In addition, Figure 6 also gives the standard deviation in the form of an error bar (a dot on the graph indicates a standard deviation of more than 100s). Figure 7 gives the number of iterations for all bundling schemes on the large instances. Table 4 also presents the number of cuts generated per iteration for each bundling scheme.

Leg bundling is the best performing scheme: It is both more efficient and more stable as the size of the instances grows. A probable explanation is that the bus leg in the trip filtering clusters trips with near-identical features with respect to their network usage (same initial and final hubs). As a consequence, the resulting cuts generate efficient partitions of the solution space. These results concur with those presented by de Camargo et al. (2008) who showed that the computational overhead of generating a cut per commodity far outweighs the potential gain in iterations and those by Contreras et al. (2011) who aggregate cuts by hubs with encouraging results.

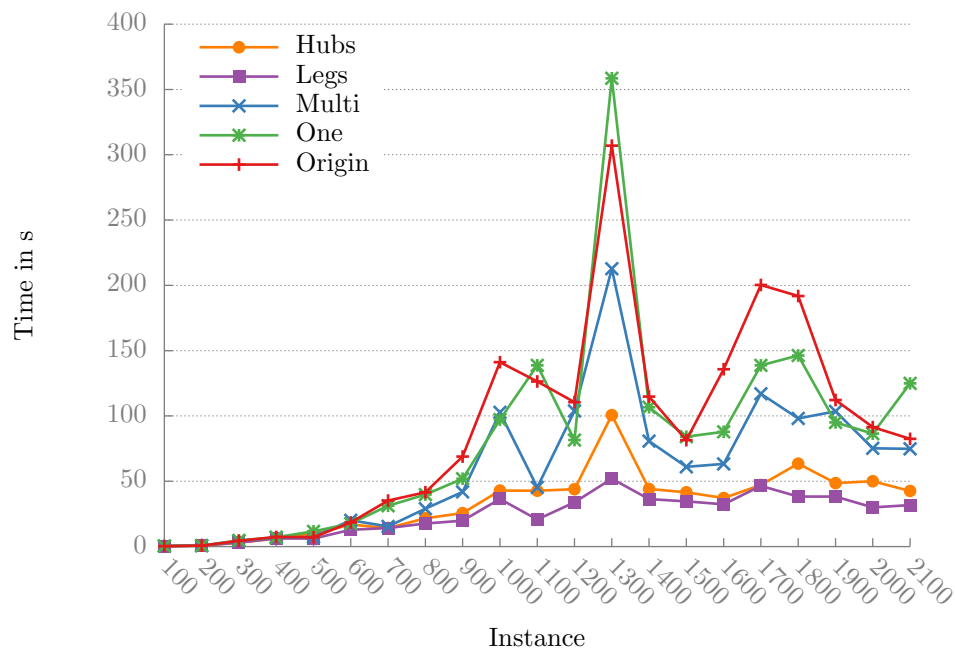


Figure 5 Comparison of Aggregation Schemes on Small Instances

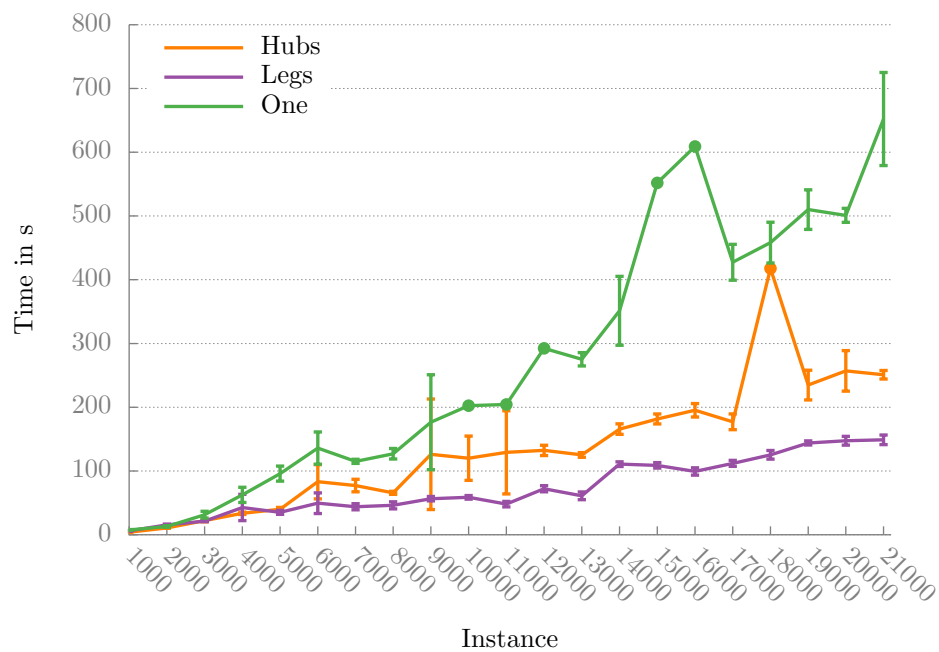


Figure 6 Comparing Cut Bundling Schemes.

7.2.3. Core Point Update Figure 8 presents the benefits of the update rule for the core point using the set of twenty potential hubs H_{20} and $\alpha = 0.1$. The key message is that the core point updates make the Benders decomposition both faster and more stable.

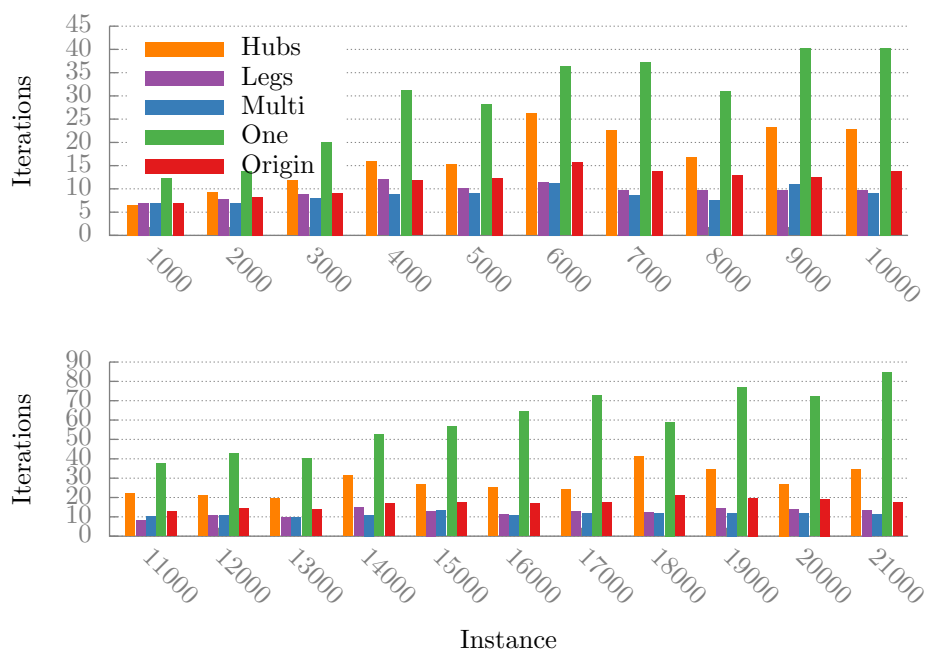


Figure 7 Number of Iterations for each Bundling Schemes on Larges Instances.

Instance	Hubs		Legs		Origin		Multi	
	H_{10}	H_{20}	H_{10}	H_{20}	H_{10}	H_{20}	H_{10}	H_{20}
1000	10	20	83	234	395	491	658	811
2000	10	20	90	280	607	716	1290	1558
3000	10	20	89	309	830	935	2002	2400
4000	10	20	89	322	962	1104	2659	3206
5000	10	20	90	328	1109	1257	3364	4036
6000	10	20	90	346	1200	1357	4036	4851
7000	10	20	90	339	1262	1427	4613	5568
8000	10	20	90	351	1360	1513	5315	6409
9000	10	20	90	354	1387	1556	6008	7182
10000	10	20	90	353	1485	1667	6741	8084
11000	10	20	90	357	1528	1712	7329	8863
12000	10	20	90	362	1582	1762	8010	9623
13000	10	20	90	360	1598	1793	8678	10447
14000	10	20	90	364	1644	1846	9290	11230
15000	10	20	90	359	1680	1872	9943	12000
16000	10	20	90	364	1721	1918	10659	12838
17000	10	20	90	364	1759	1953	11374	13665
18000	10	20	90	366	1781	1988	12006	14446
19000	10	20	90	369	1800	2002	12651	15236
20000	10	20	90	369	1828	2032	13310	16033
21000	10	20	90	369	1853	2054	13968	16824

Table 4 Number of Cuts per Iteration for the Various Bundling Schemes.

7.2.4. MIP versus Benders Decomposition Figure 9 compares the standard MIP model against the final version of our Benders decomposition algorithm (independent Pareto sub-

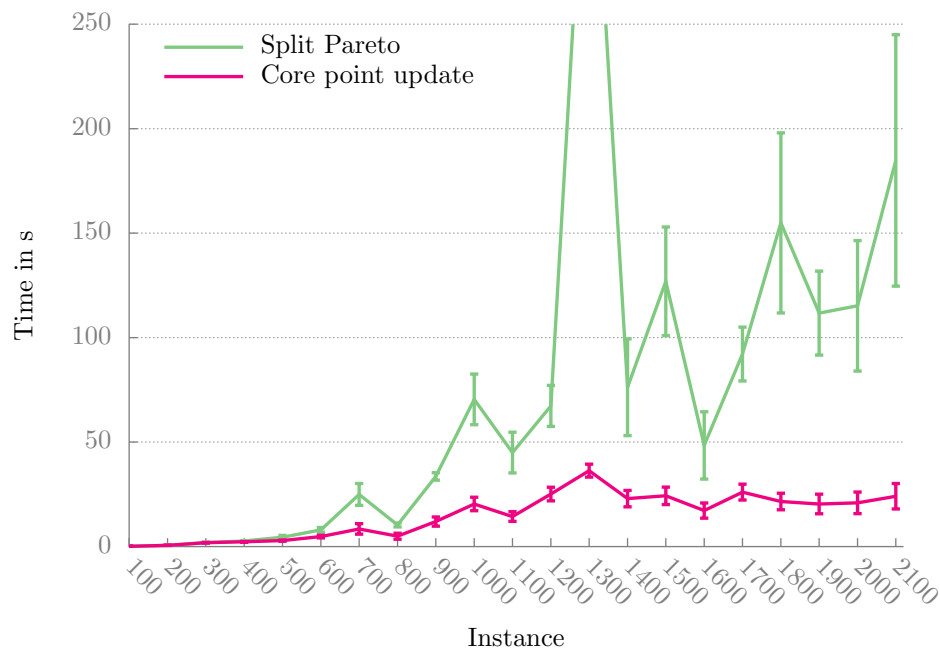


Figure 8 The Impact of the Core Point Updating Rule using H_{20} & $\alpha = 0.1$.

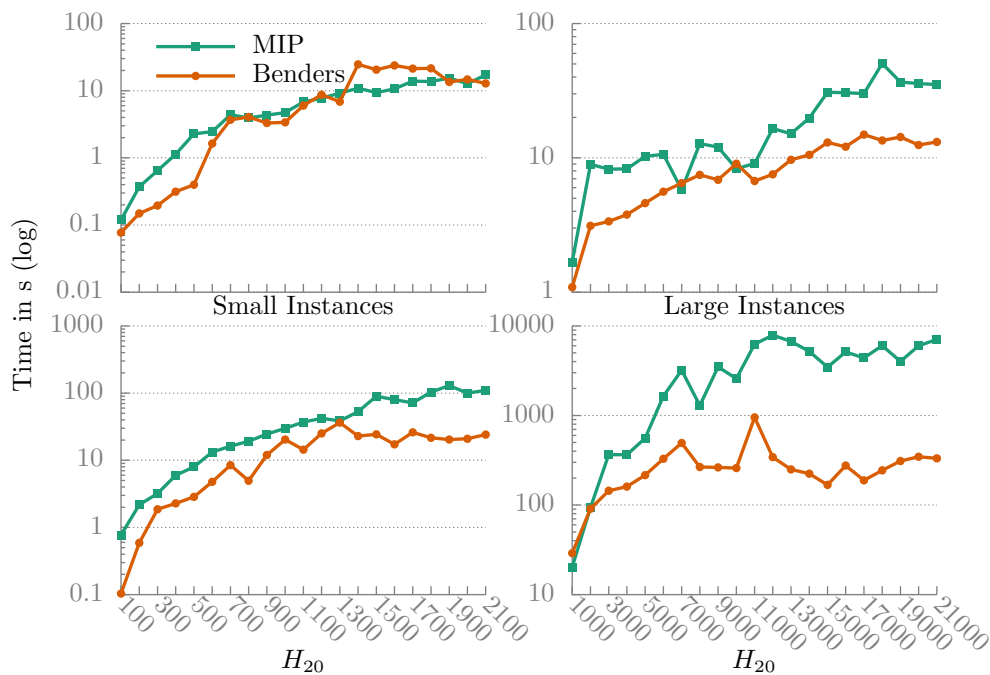


Figure 9 Comparing the MIP Model and the Final Benders Decomposition.

problems, leg aggregation, core point update). The experiments use $\alpha = 0.1$ as it gives the most *realistic* results. The first column presents the results on the set of small instances and the second column on the large instances. The first line uses a configuration with ten hubs (H_{10}) and the second line twenty hubs (H_{20}). As the instances grow larger, the gap between the standard MIP

Day	BUSPLUS			ACTION	
	Buses (\$)	Cost (\$)	Time (s)	Cost (\$)	Time (s)
Monday	45728.57	369420.37	829.88	402006.75	1635.22
Tuesday	45728.57	362746.82	828.77	402006.75	1635.10
Wednesday	46436.58	372214.03	828.55	402006.75	1620.79
Thursday	45899.13	376147.06	825.21	402006.75	1632.79
Friday	43893.83	350709.85	819.64	402006.75	1610.85

Table 5 Time and Cost Comparison Between BusPlus and Action.

and the Benders decomposition becomes increasingly pronounced: Benders decomposition is almost two orders of magnitude faster on the largest configuration.

7.3. Benefits on the Case Study

We conclude this section by evaluating the approach on the real case-study: The public transit system in Canberra. Table 5 compares the results of our BUSPLUS approach against the current public transportation system of Canberra known as ACTION. For BUSPLUS, the table reports the cost of the bus network, the total cost of the system, and the average travel time in the three columns. For ACTION, the table reports the cost of the system and the average travel time. The experimental settings differ slightly between BUSPLUS and ACTION to reflect the state of the system as accurately as possible. In particular, we do not include the waiting time for boarding buses in ACTION, since this data is not available. We also set α to 0.1 as it seems to give the most realistic networks in terms of the balance between costs and convenience.

The results show that the BUSPLUS approach divides the average travel time by two, even when taking into account the bus waiting time. The main reason for this reduction is the significant number of trips being served by taxis only. Moreover, even though the main expense of BUSPLUS comes from the taxi trips, the overall cost of the system is about ten percent lower than the cost of ACTION. This is an interesting result, since the taxi costs here are intentionally exaggerated and correspond to existing full fares.

Figure 10 displays the networks obtained for the H_{10} and H_{20} configurations. The H_{20} configuration is particularly interesting: It is organized around two interconnected centers, which serve some of the most densely populated areas with short, flower-like routes.

8. Conclusion

The BUSPLUS project aims at improving the off-peak hours public transit service in Canberra, Australia. The city covers a wide geographical area which makes public transportation particularly challenging. To address the difficulty, the BUSPLUS project proposes a hub and shuttle model consisting of a combination of a few high-frequency bus routes between key hubs and a large

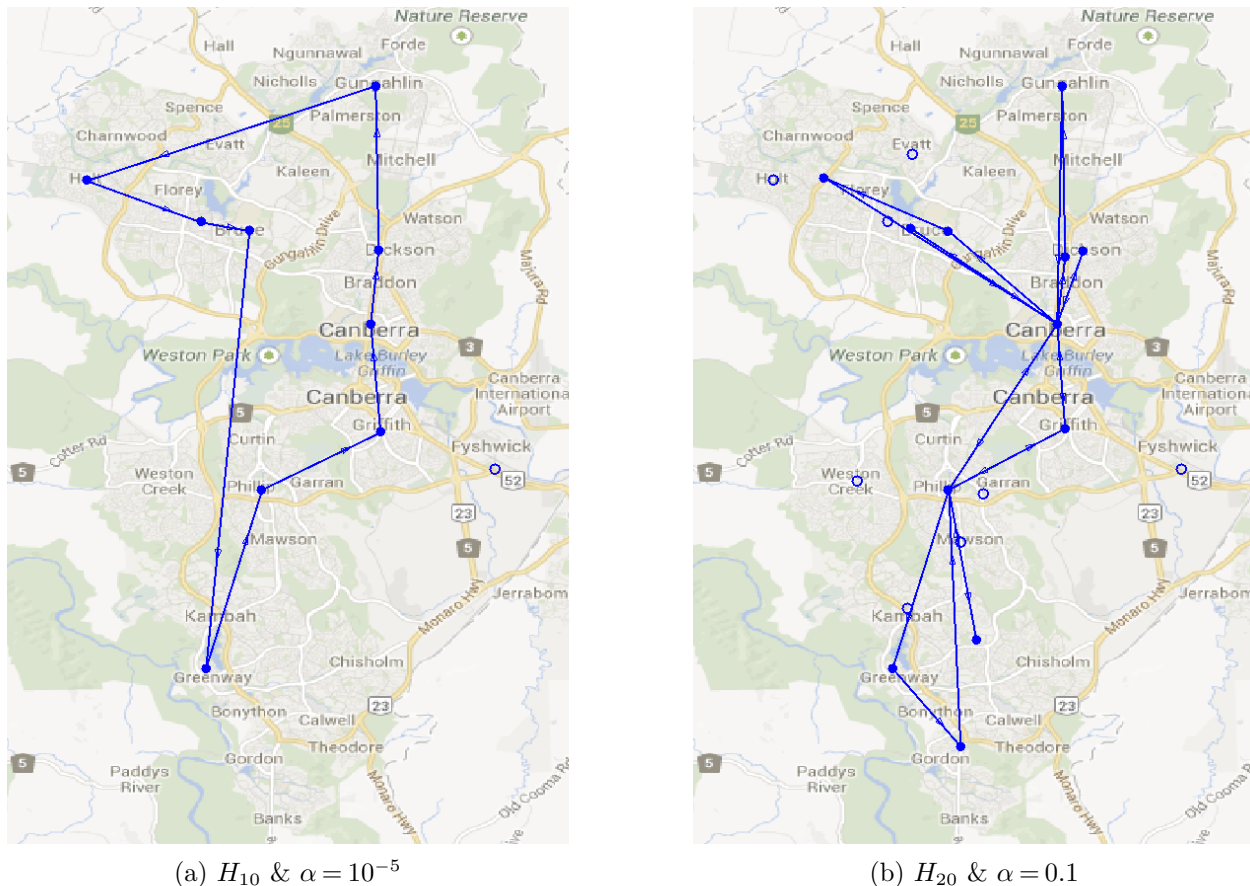


Figure 10 Example networks using geographical data

number of shuttles that bring passengers from their origin to the closest hub and take them from their last bus stop to their destination.

This paper focused on the design of bus network, which can be modeled as a variant of the hub-arc location problem. It proposed a number of pre-processing techniques, trip filtering and link filtering, to reduce the problem size. Then it introduced a Benders decomposition approach that uses dedicated solution techniques for solving independent sub-problems, Pareto optimal cuts, cut bundling, and core point update. The benefits of these design decisions are validated using real-world data from public transit system in Canberra. The Benders decomposition outperforms the natural MIP formulation by two orders of magnitude. Moreover, the paper showed that the hub and shuttle model may decrease transit time by a factor of 2, while staying within the costs of the existing transit system.

There are several directions for future work, including the use of parallel computation for the subproblems, the merging of the hub selection and the network design in a single optimization problem, and the ability to take into account of the scheduling, routing, and fleet sizing aspects.

It is also useful to mention that the BUSPLUS project will undergo live trial in the near future.

References

- Ahuja, Ravindra K, Thomas L Magnanti, James B Orlin. 1988. Network flows. Tech. rep., DTIC Document.
- Benders, J. F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4**(1) 238–252. doi:10.1007/BF01386316.
- Birge, John R, Francois V Louveaux. 1988. A multicut algorithm for two-stage stochastic linear programs. *European Journal of Operational Research* **34**(3) 384–392.
- Campbell, J. F., a. T. Ernst, M. Krishnamoorthy. 2005a. Hub Arc Location Problems: Part II—Formulations and Optimal Algorithms. *Management Science* **51**(10) 1556–1571. doi:10.1287/mnsc.1050.0407. URL <http://pubsonline.informs.org/doi/abs/10.1287/mnsc.1050.0407>.
- Campbell, J. F., a. T. Ernst, M. Krishnamoorthy. 2005b. Hub Arc Location Problems: Part I—Introduction and Results. *Management Science* **51**(10) 1540–1555. doi:10.1287/mnsc.1050.0406. URL <http://pubsonline.informs.org/doi/abs/10.1287/mnsc.1050.0406>.
- Contreras, Ivan, Jean-François Cordeau, Gilbert Laporte. 2011. Benders decomposition for large-scale uncapacitated hub location. *Operations research* **59**(6) 1477–1490.
- de Camargo, R. S., G. Miranda, H. P. Luna. 2008. Benders decomposition for the uncapacitated multiple allocation hub location problem. *Computers and Operations Research* **35**(4) 1047–1064. doi:10.1016/j.cor.2006.07.002.
- de Sá, Elisangela Martins, I. Contreras, Jean-Francois Cordeau. 2015a. Exact and heuristic algorithms for the design of hub networks with multiple lines. *European Journal of Operational Research* **246**(1) 186–198.
- de Sá, Elisangela Martins, Ivan Contreras, Jean-François Cordeau, Ricardo Saraiva de Camargo, Gilberto de Miranda. 2015b. The Hub Line Location Problem. *Transportation Science* **49**(3) 500–518. doi:10.1287/trsc.2014.0576. URL <http://pubsonline.informs.org/doi/10.1287/trsc.2014.0576>.
- de Sá, Elisangela Martins, Ricardo Saraiva de Camargo, Gilberto de Miranda. 2013. An improved Benders decomposition algorithm for the tree of hubs location problem. *European Journal of Operational Research* **226**(2) 185–202.
- Geoffrion, Arthur M, Glenn W Graves. 1974. Multicommodity distribution system design by Benders decomposition. *Management science* **20**(5) 822–844.
- Gurobi Optimization, Inc. 2015. Gurobi optimizer reference manual. URL <http://www.gurobi.com>.
- Hakimi, S Louis. 1964. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations research* **12**(3) 450–459.
- Labbé, Martine, Hande Yaman. 2008. Solving the hub location problem in a star–star network. *Networks* **51**(1) 19–33.
- Magnanti, Thomas L, Richard T Wong. 1981. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research* **29**(3) 464–484.

- Marín, Ángel G., Patricia Jaramillo. 2009. Urban rapid transit network design: Accelerated Benders decomposition. *Annals of Operations Research* **169**(1) 35–53. doi:10.1007/s10479-008-0388-0.
- Papadakos, Nikolaos. 2009. Integrated airline scheduling. *Computers & Operations Research* **36**(1) 176–195.
- Yaman, Hande. 2008. Star p-hub median problem with modular arc capacities. *Computers & Operations Research* **35**(9) 3009–3019.