



A column generation algorithm for tactical timber transportation planning

Gregory Rix^{1,2*}, Louis-Martin Rousseau^{1,2} and Gilles Pesant^{1,2}

¹Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Montreal, Canada; and ²École Polytechnique de Montréal, Montreal, Canada

We present a tactical wood flow model that appears in the context of the Canadian forestry industry, and describe the implementation of a decision support system created for use by an industrial partner. In this problem, mill demands and harvested volumes of a heterogeneous set of log types are given over a multi-period planning horizon. Wood can be stored at the forest roadside before delivery at a financial cost. Rather than solve this as a network linear programme on the basis of out-and-back deliveries, we choose to model this problem as a generalization of a log-truck scheduling problem. By routing and scheduling the trucks in the resolution, this allows us to both anticipate potential backhaul opportunities for cost and fuel savings, and also minimize queuing times at log-loaders, management of which is a major concern in the industry. We model this problem as a mixed integer linear programme and solve it via column generation. The methodology is tested on several case studies.

Journal of the Operational Research Society advance online publication, 15 January 2014; doi:10.1057/jors.2013.170

Keywords: forestry; vehicle routing; decision support system; integer programming; column generation

Introduction

According to *Natural Resources Canada* NRC (2011), Canada has 402.1 million hectares of forest and other wooded land, making up approximately 10% of the world's forest cover. It is therefore unsurprising that Canada is the world's largest exporter of forest products: in 2008, the value of all exports from this industry was over US\$30 billion. Overall, the forest industry accounts for approximately 2% of national gross domestic product. When dealing with an economy of this scale, it is clear that performing all operations as efficiently as possible can lead to tremendous financial savings. Therefore, it is a necessity that the use of optimization models and methods are commonplace in this industry, both for the obvious economic and also environmental reasons.

Because of the geographical nature of Quebec, transportation accounts for approximately 30% of the total cost of wood. The average distance between the forest where the wood is collected and the mills to which it is delivered is approximately 150 km. However, as significant as this aspect of the supply chain is, most Canadian forest companies have a planner derive the truck schedules manually.

This is changing, and recent emphasis has been put on the development of planning methods and decision support systems in this domain to take advantage of potential savings

(Audy *et al.*, 2012). We note three main phases of transportation planning that are focused on in this paper: allocation, routing, and scheduling. Allocation involves making optimal allocations of supply points to demand points in order to minimize hauling distance. Routing involves creating routes for the trucks that allow us to take advantage of backhaul opportunities. Recent advancements in the use of multi-product truck trailers have generated new opportunities in this regard. Finally, scheduling is important in this industry because of unproductive but still expensive truck and driver waiting time (queuing) when coordinating with loading equipment, a problem that is more unique to the forestry industry.

Wood allocation decisions are commonly made at the tactical level of planning, for example, when formulating annual harvest and wood flow schedules. When these decisions are already determined, there is much less flexibility when determining truck schedules in operational planning, and perhaps further minimization could be achieved through smarter tactical decisions. It is for this reason that we create a tactical wood flow model that prioritizes routing and scheduling, allowing us to realize additional cost savings.

Thus, the contribution of this paper is to define a tactical transportation planning problem over a 1-year time horizon that incorporates log-truck scheduling elements in order to more accurately minimize transportation costs and queuing times. We then develop a column generation methodology to apply to this problem and give computational results on several industrial case studies. In addition, we describe the implementation into a decision support system that is in use by an industrial partner.

*Correspondence: Gregory Rix, Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation, CP 6079, succ Centre-Ville, Québec, Montreal, H3C3A7, Canada.
E-mail: greg.rix@polymtl.ca

In the next section, we discuss related work. In the sections ‘Problem definition’ and ‘Mathematical formulation’, we define the problem. In the section ‘Methodology’, we discuss our methodology. In the sections ‘Case studies’ and ‘Experimental results’, we discuss the case studies that motivated this problem and the computational results. The section ‘Conclusion and future work’ concludes this paper.

Literature review

An introduction to optimization in the forestry industry is given in Rönnqvist (2003). The author distinguishes between the strategic (over 5 years), tactical (between 6 months and 5 years), and operational (between 1 day and 6 months) planning levels, though exact planning horizons used by planners are related to the context of their country and company. We will focus on the tactical and operational planning horizons in this paper.

Tactical planning horizon

At the tactical level of wood procurement planning, little emphasis is placed on transportation. Along with the upgrade of transportation infrastructures and adjustment of transportation equipment capacity, volume allocations from supply points to demand points are often decided. When formulating annual harvest plans, the total cost of wood flow from forest sites to mills is minimized based on out-and-back travel distance, with more detailed transportation plans determined at the operational level. Epstein *et al* (2007) show how this can be modelled as a simple network flow problem. These constraints are typically incorporated in harvest planning. Rönnqvist (2003) gives a simple model in which the objective function measures two costs: the cost of harvesting a forest area by a crew in a specific period and the cost of delivering each unit of wood from a forest site to a mill.

This harvest planning can be generalized to more detailed models considering other constraints. One of these constraints that is most common is the determination of wood storage at terminals, and this can be seen in many tactical models (Karlsson *et al*, 2004; Gunnarsson *et al*, 2007). Gémieux (2009) gives a recent survey on tactical wood procurement planning and outlines a mixed integer programming (MIP) model that represents the context in Quebec.

In a preliminary version of the methodology reported in this paper (Rix *et al*, 2011), a routing-based tactical wood flow model was developed and a column generation procedure was applied. Owing to optimization of backhaul opportunities including routing variables in the formulation yielded additional savings: up to 5% was reported. By taking the routing decisions into account in a multi-period tactical plan, this problem generalizes the well-studied inventory routing problem (Coelho *et al*, 2012).

Operational planning horizon

At the operational level, more detailed transportation plans are determined where more complex routes are constructed under the objective of minimizing transportation costs. When necessary, synchronization with log-loaders at supply and demand points are scheduled and queuing times are minimized. In the forestry industry, the problem of creating these plans is commonly called the log-truck scheduling problem (LTSP).

Transportation decisions are of course not unique to the forestry industry, and the LTSP is a generalization of the well-studied pickup-and-delivery problem (PDP) (Berbeglia *et al*, 2007). Berbeglia *et al* (2007) distinguish between three structures of PDP: one-to-many-to-one, many-to-many, and one-to-one. The LTSP can be classified into the latter two structures depending on whether supply and demand points are unpaired or paired, respectively, and this paper will consider both variations. In addition, most PDPs seen in the literature have each customer serviced by exactly one vehicle; however, in the forestry industry, a single visit by a single vehicle is almost always insufficient to transport the entire demand. Therefore, multiple visits are required by either one or many vehicles. In the literature, problems with this feature are referred to as split load vehicle routing problems (Dror *et al*, 1994).

There is much literature devoted specifically to the LTSP, and for a recent survey we direct the reader to Audy *et al* (2012). In Gronalt and Hirsch, (2007), the Unified Tabu Search Algorithm (Cordeau *et al*, 2001) was applied to the one-to-one LTSP. Column generation has also been applied to the LTSP (Palmgren *et al*, 2004; Rey *et al*, 2009). Palmgren *et al* (2004) solved the subproblem using a k -shortest path algorithm and a branch-and-price heuristic was used to find integer feasible solutions, while Rey *et al* (2009) utilized dynamic programming in the subproblem and branch-and-bound was used after solving the relaxation with column generation.

Several decision support tools have been developed. EPO (Linnainmaa *et al*, 1995) is a knowledge-based system developed in Finland in the 1990s that outputs a weekly schedule for each truck. MaxTour (Gingras *et al*, 2007) is a planning method incorporated in the forestry operations control platform FPSuite (n.d.), developed by FPIinnovations, that optimizes backhaul opportunities in transportation planning.

However, very few approaches seen in the literature take into account synchronization with log-loaders, which can be a crucial aspect in practice. A weekly problem is considered in El Hachemi *et al* (2013), in which the authors assign forest supply to mills by solving a tactical MIP, and they then use both constraint programming and constraint-based local search to schedule the trucks afterwards. The same two-phase approach has been adopted, with the scheduling phase modelled as a mixed integer linear programme using a flow-based formulation (El Hachemi *et al*, 2011).

Problem definition

The problem we consider in this paper is a tactical wood flow model. We assume an annual harvest plan as input, and we will solve a single tactical model of determining the wood flow and storage, while taking into account truck routing and scheduling decisions over the planning horizon. This allows for smarter tactical decisions to be made, through taking advantage of savings generated through optimization of all of allocation, routing, and scheduling. While the scenario may change over the year and reoptimization may be necessary on a rolling basis, this still gives a plan for the present that improves the operational planning process in the future.

We assume a multi-period planning horizon, with each period partitioned into a set of working days. We are next given a harvest plan of log assortments at forest sites over this horizon, with these multiple assortments differing in terms of species, quality, length, and/or diameter. While these are anticipated future volumes, advancements in scanning of forest stands make these estimates accurate enough for planning purposes. We also assume non-constant but deterministic mill demands over the horizon, though reoptimization may be necessary as demands change over time. Multiple classes of trucks are available to transport the wood, each with a different capacity and loading requirements. We note that different trucks with different capacities implies that volumes must be expressed in cubic metres (m^3), rather than simply truckloads as is common in the current literature.

Under the current industrial constraints, some trucks are equipped with onboard loaders and thus do not need to synchronize with a separate loader at the forest; however, every truck must unload its capacity via the equipment based at the mill. Whenever a truck must synchronize with a loader at the forest or mill, if the loader is busy when the truck arrives then the truck must wait, yielding unproductive labour costs. While we do not take into account loader routing or minimization of loader operating costs, we do place an upper bound, per period, on the number of forest sites a loader can be assigned to. Each loader can then only move around within this site, movement that is beyond the scope of this model. By bounding the number of ‘active’ forest sites, we will assist future decision makers by simplifying the operational loader scheduling.

This LTSP is then to determine the quantity of each assortment to deliver from each forest to each mill in each period, the necessary storage between periods at each forest and each mill, the traversal of log-truck routes in order to deliver the produced logs to the mills, and the assignment of log-loaders to forest sites throughout the horizon.

In addition to satisfying mill demands and not exceeding the harvested quantities, several other constraints must be satisfied. There exists a given limit on the length of time wood can remain in the forest. This may vary by period, as wood deteriorates less quickly in the winter than in other seasons. With respect to routing, if a truck is to be used on a specific day, it must begin and end its day at the same mill. The truck then

alternates between forest sites where it loads wood, and mills where it then unloads. Truck capacity cannot be exceeded, and the truck cannot carry different assortments at the same time. In addition, each mill has operating hours over which it can receive wood. A truck can leave a mill before it opens and return to a mill after it closes; however, it will have to wait until the mill is open to unload its load.

A final constraint that is often used is important for planning truck fleet size and satisfying labour constraints: we also must ensure a balanced schedule. That is, each day a minimum and maximum number of daily truck routes must be assigned that do not deviate too far from the mean over the horizon. However, if the inputs of the model are a harvest plan with a large standard deviation of quantity produced per period, these constraints cannot be enforced as there must naturally be a correlation between the number of cubic metres to haul and the number of trucks on the road.

Mathematical formulation

Let us define F to be the set of forest sites, M to be the set of mills, L to be the set of log assortments, T to be the set of truck classes, P to be the set of periods in the planning horizon, and D_p to be the set of all working days in each period p . Moreover, we must discretize the time dimension of each day into $I = \{i_0, i_1, \dots, i_n\}$, with δ denoting the interval duration between any consecutive i . We then let J be the set of all feasible log-truck routes, every activity of which (driving, (un) loading, and waiting for a loader) has its duration approximated by some multiple of δ . For notational purposes, we partition J by the truck class t and by the mill m at which the route originates and terminates, into J_t , J_m , and J_{mt} . The cost of a given route is then easily calculated as a function of per hour operating and waiting costs.

We now define the input data of the model:

h_{flp}	= the quantity of assortment l harvested at forest f in period p ,
d_{mlp}	= the demand of assortment l at mill m in period p ,
i_{fl}^F	= the initial inventory of assortment l at forest f ,
i_{ml}^M	= the initial inventory of assortment l at mill m ,
w_{flp}	= the maximum storage time at forest f of assortment l harvested in period p ,
$w_{flpp'}$	= $1(p' + w_{flp'} \geq p)$ = binary parameter indicating if harvest of assortment l in forest f in period $p' \leq p$ can still remain at roadside in p ,
k_{tl}	= the capacity of assortment l of truck class t ,
v_t	= $\begin{cases} 1 & \text{Truck class } t \text{ requires synchronization} \\ & \text{with a loader at the forest,} \\ 0 & \text{Truck class } t \text{ is selfloading,} \end{cases}$
c_{flp}^F	= the per unit storage of assortment l at forest f in period p ,
c_{mlp}^M	= the per unit storage of assortment l at mill m in period p ,

c_j^J = the cost of route j ,
 c_t^T = the fixed cost of operating a truck of type t on a working day,
 a_{fmlj} = the number of trips from forest f to mill m carrying assortment l on route j ,
 L_{jfi} = $\begin{cases} 1 & \text{A truck traversing route } j \text{ is loading wood at forest } f \text{ over interval } i, \\ 0 & \text{Otherwise,} \end{cases}$
 U_{jmi} = $\begin{cases} 1 & \text{A truck traversing route } j \text{ is unloading wood at mill } m \text{ over interval } i, \\ 0 & \text{Otherwise.} \end{cases}$
 n_{mpt}^T = the number of available trucks of class t based at mill m in period p ,
 $\varepsilon \in [0, 1]$ = the allowable deviation per period from the mean number of routes,
 n_p^L = the number of loader available in period p to assign to forest sites,
 n_{mp}^L = the number of loaders available in period p at mill m .

The variables of the model are given below:

y_{jpd} = the number of times route j is traversed on day d in period p ,
 T_{mpdt} = the number of daily truck routes based at mill m in period p using truck class t
 z_{mlp}^M = the quantity of assortment l stored at mill m entering period $p \leq |P| + 1$,
 z_{fp}^F = the quantity of assortment l stored at forest f entering period $p \leq |P| + 1$,
 x_{fmlpt} = the quantity of assortment l delivered from forest f to mill m using truck class t in period p ,
 L_{fp} = $\begin{cases} 1 & \text{A loader is assigned to forest } f \text{ in period } p, \\ 0 & \text{Otherwise.} \end{cases}$

The problem can then be formulated as the minimization of the objective function:

$$\sum_{p \in P} \sum_{d \in D_p} \sum_{j \in J} c_j^J y_{jpd} + \sum_{m \in M} \sum_{p \in P} \sum_{d \in D_p} \sum_{t \in T} c_t^T T_{mpdt} + \sum_{f \in F} \sum_{l \in L} \sum_{p \in P} c_{flp}^F z_{fp}^F + \sum_{m \in M} \sum_{l \in L} \sum_{p \in P} c_{mlp}^M z_{mlp}^M \quad (1)$$

subject to:

$$z_{ml0}^M = i_{ml}^M, \forall m \in M, l \in L, \quad (2)$$

$$z_{f0}^F = i_{f0}^F, \forall f \in F, l \in L, \quad (3)$$

$$z_{mlp}^M + \sum_{f \in F} \sum_{t \in T} x_{fmlpt} - d_{mlp} = z_{ml(p+1)}^M, \quad \forall m \in M, l \in L, p \in P, \quad (4)$$

$$z_{flp}^F + h_{flp} - \sum_{m \in M} \sum_{t \in T} x_{fmlpt} = z_{fl(p+1)}^F, \quad \forall f \in F, l \in L, p \in P, \quad (5)$$

$$\sum_{p'=0}^p w_{flpp'} h_{flp'} \geq z_{flp}^F, \forall f \in F, l \in L, p \leq |P| + 1, \quad (6)$$

$$\sum_{j \in J_i} k_{it} a_{fmlj} \sum_{d \in D_p} y_{jpd} \geq x_{fmlpt}, \forall f \in F, m \in M, l \in L, p \in P, t \in T, \quad (7)$$

$$\sum_{j \in J_m} y_{jpd} = T_{mpdt}, \forall m \in M, p \in P, d \in D_p, t \in T \quad (8)$$

$$T_{mpdt} \leq n_{mpt}^T, \forall m \in M, p \in P, d \in D_p, t \in T \quad (9)$$

$$\sum_{m \in M} \sum_{t \in T} T_{mpdt} \geq \frac{1-\varepsilon}{\sum_{p' \in P} |D_{p'}|} \sum_{m \in M} \sum_{p' \in P} \sum_{d' \in D_{p'}} \sum_{t \in T} T_{mpdt}, \quad \forall p \in P, d \in D_p, \quad (10)$$

$$\sum_{m \in M} \sum_{t \in T} T_{mpdt} \leq \frac{1+\varepsilon}{\sum_{p' \in P} |D_{p'}|} \sum_{m \in M} \sum_{p' \in P} \sum_{d' \in D_{p'}} \sum_{t \in T} T_{mpdt}, \quad \forall p \in P, d \in D_p, \quad (11)$$

$$\sum_{f \in F} L_{fp} \leq n_p^L, \forall p \in P, \quad (12)$$

$$\sum_{j \in J_i} U_{jmi} y_{jpd} \leq n_{mp}^L, \forall m \in M, p \in P, d \in D_p, i \in I, \quad (13)$$

$$\sum_{t \in T} v_t \sum_{j \in J} L_{jfi} y_{jpd} \leq L_{fp}, \forall f \in F, p \in P, d \in D_p, i \in I, \quad (14a)$$

$$\sum_{t \in T} v_t \sum_{d \in D_p} \sum_{m \in M} \sum_{l \in L} x_{fmlpt} \leq \Omega L_{fp}, \forall f \in F, p \in P, \quad (15a)$$

$$\sum_{m \in M} \sum_{j \in J_m} \sum_{i \in I} U_{jmi} y_{jpd} \leq \sum_{m \in M} \sum_{j \in J_m} \sum_{i \in I} U_{jmi} y_{jp,d+1}, \quad (16)$$

$$\begin{aligned} \forall p \in P, d \leq |D_p| - 1, \\ L_{fp} \in \{0, 1\}, \forall f \in F, p \in P. \end{aligned} \quad (17)$$

$$\begin{aligned} y_{jpd}, T_{mpdt} \in \mathbb{Z}^+, \forall m \in M, j \in J_m, \\ p \in P, d \in D_p, t \in T. \end{aligned} \quad (18)$$

$$\begin{aligned} z_{mlp}^N, z_{jlp}, x_{jmlp} \in \mathbb{R}^+, \forall f \in F, \\ m \in M, l \in L, p \in P. \end{aligned} \quad (19)$$

We denote this problem by (P) . The objective function (1) minimizes total costs associated with driving and storage. Constraints (2) and (3) set the initial inventories at the mills and forests, respectively. Constraints (4) and (5) link the storage variables of successive periods at the mills and forests, respectively. The non-negativity of all variables ensure that forest supply and mill demands are respected. Constraint (6) ensures that wood is not left at the forest longer than allowed by forcing the volume in storage in period p to be bounded above by the volume of any prior harvest that is still fresh in p . We note that for these constraints to be valid, a first in first out system must be utilized by the drivers, meaning that a driver will always pick up the oldest wood of the specified log assortment.

Constraint (7) forces the quantity delivered to respect the capacities of all trucks making that trip. Constraint (8) fixes the number of routes originating from each mill in each period, and Constraint (9) binds this by the associated availability. Constraints (10) and (11) ensure a balanced schedule in terms of the number of truck routes traversed at every day of the horizon by binding it by both the allowed percentage above $(1 + \epsilon)$ and below $(1 - \epsilon)$ the mean number of routes per day.

Constraint (12) limits the total number of loaders assigned to forests in a period. Constraints (13) and (14a) assign each loader to only one truck at any time. Constraint (15a) comprises redundant constraints that force a loader to be assigned to a forest in any period in which a truck requires one, with Ω a sufficiently large constant. Constraint (16) breaks the symmetry between the days that define a period by having the number of trucks unloaded at any mill always be an increasing function over the days of a given period. We note that we are not truly enforcing this monotonicity, as the days of a period are arbitrary and can be permuted without loss of generality if required. Constraint (17) forces the loader-to-forest assignment variables to be binary. Finally, Constraints (18) and (19) enforce the non-negativity of the other variables, as well as discretize those that count log-truck routes. We denote by \mathbb{Z}^+ and \mathbb{R}^+ the sets of non-negative integers and reals, respectively.

We note that, in cases where the truck fleet is homogeneous, we will express volumes in truckloads as is more common in LTSP literature. The only necessary change to the model in this case is setting the parameter k_{il} to be equal to 1. In the model as

presented, we do note that the volume loaded at every forest on every route is not explicitly defined. However, this can easily be determined from a feasible solution in a greedy fashion. This is, of course, done to dramatically reduce the size of the formulation.

Finally, in many cases in the Canadian forestry industry, there exist contractual obligations that stipulate what proportion of the harvest at each forest site must be sent to each mill. This problem then has more similarities to classic PDPs. We can represent this in our mathematical model by duplicating our assortments into a different assortment for every mill that has a demand of that assortment. We then treat each forest site f as a vector of multiple forests $\vec{f} = (f_{m_1}, f_{m_2}, \dots, f_{m_k})$ for all mills that the forest must serve, and treat each of these as an individual forest throughout the model that provides the needed quantity of the assortments that correspond to that mill. The only change in the model is to then associate vector \vec{f} with a single vector of loader variables $(L_{\vec{f}p})_{p \in P}$, and replace Constraints (14a) and (15a) as follows:

$$\begin{aligned} \sum_{f \in \vec{f}} \sum_{t \in T} v_t \sum_{j \in J} L_{jfi} y_{jpd} \leq L_{\vec{f}p}, \forall \vec{f} \in F, p \in P, \\ d \in D_p, i \in I, \end{aligned} \quad (14b)$$

$$\sum_{f \in \vec{f}} \sum_{t \in T} v_t \sum_{d \in D_p} \sum_{m \in M} \sum_{l \in L} x_{jmlpt} \leq \Omega L_{\vec{f}p}, \forall \vec{f} \in F, p \in P. \quad (15b)$$

Methodology

We propose to apply column generation to this problem, similar to the method used by Rix *et al.* (2011). We must generalize the model to take into account multiple periods, loader synchronization, and several other constraints.

Initial restricted problem

To solve the linear relaxation of (P) via column generation, we must first determine an initial set of columns J' that makes the problem feasible. As this may be very difficult, we propose to relax the Constraints (13) and (14) associated with loader synchronization and penalize any violation in the objective function. With these constraints relaxed, the most intuitive and simple route sub set of J that guarantees to make the problem feasible is the set of out-and-back routes defined by (f, m, l) for f, m, l in F, M, L , respectively. We repeat this trip as many times as time windows will allow, and add this route as a column for each day of the planning horizon in which this trip is valid for the harvest plan and mill demands of the current period. We denote this restricted master problem (P') .

After solving the linear programming (LP) relaxation of (P') , we store the dual values associated with Constraints (7), (8), (13), (14), and (16), which we denote π_{jmlp} , λ_{mpdt} , σ_{mpdt}^U , σ_{jpd}^L , and γ_{pd} , respectively. Our search then begins for negative

reduced cost columns with which to enrich the model to improve the objective value of the optimal solution. We propose to find these columns by performing a set of dynamic programming algorithms, one for each day d of each period p , and for each truck class t .

Enriching the model with column generation

To solve these subproblems, we must first construct a space-time network, which we denote $G = (V, A)$, for the given day of the period and truck class. In this space-time network, we discretize the time dimension of the day as in the problem definition, letting δ again be the interval length between two consecutive timesteps.

We define the network with vertex set $V = (M \cup F) \times I \times \{0, 1\}$, where 0 corresponds to having an empty truck at that location and 1 corresponds to having a loaded truck. The arc set is then $A = A_w \cup A_u \cup A_l \cup A_{ud} \cup A_{ld}$, where A_w , A_u , A_l , A_{ud} , and A_{ld} represent waiting, loading, unloading, unloaded driving, and loaded driving arcs, respectively. The cost of each arc c_{uv} is then easily calculated as a function of per hour operating and waiting costs of that truck. However, in calculating the reduced cost of a route, we modify these arc costs as follows, letting i_{-uv} and \bar{i}_{uv} denote the first and last time intervals over which the arc intersects:

$$c_{uv} \leftarrow \begin{cases} c_{uv} & (u, v) \in A_w, \\ c_{uv} - \sum_{i=i_{-uv}}^{\bar{i}_{uv}} \sigma_{mpdi}^U & (u, v) \in A_u, \\ c_{uv} - \sum_{i=i_{-uv}}^{\bar{i}_{uv}} \sigma_{fpdi}^U & (u, v) \in A_l, \\ c_{uv} & (u, v) \in A_{ud}, \\ \min_{l \in L} \{ \delta d(f, m) c_{uv} - k_l \pi fmlp \} & (u, v) \in A_{ld}. \end{cases}$$

Any feasible route can then be expressed as a path in this network between any two vertices representing (unloaded) home mill m at different times in the day. The reduced cost of this route will then be equal to the cost of the path minus the dual value λ_{mpdt} , which measures a fixed reduced cost associated with operating that truck that day.

We note that this network has a clear topological ordering, which is a chronological ordering with ties broken arbitrarily. To find negative reduced cost routes to add to the master problem, we can therefore utilize a standard label-setting algorithm (Cormen *et al*, 1990, Sec 24.2) in which we associate with each vertex v a label $[pred_v, RC_v]$ that denotes the predecessor of v and the length (reduced cost) of the shortest path to v . For any day d of period p , truck class t and mill m , we provide the algorithm in the Appendix.

Thus, at every master iteration, we store the dual values of Constraints (7), (8), (13), (14), and (16), and then solve $M|I|I \Sigma_{p \in P} |D_p|$ subproblems. All negative reduced cost routes are stored and the columns are added to the master problem. We

iterate through this process until no negative reduced cost routes remain or another stopping criteria (such as a time or column limit) is reached.

Column pool management

At each iteration, upon the resolution of all subproblems, the most general method would be to add all negative reduced cost columns found to the LP. However, many of these routes will prove unnecessary and remain non-basic until the algorithm terminates. Therefore, we utilize two methods to control the size of the column pool. First, at each iteration we simply added the best (most negative reduced cost) 200 columns found. Second, for each column, we track the number of iterations for which it has been non-basic. Once a column has been inactive for 30 iterations, we delete this column from the pool and model.

Generating an integer solution

We then restore integrality to the variables y and T and re-enforce the synchronization constraints as hard constraints to generate a fully feasible solution. We then solve the resulting MIP model with the current column pool of routes via branch-and-bound to get an integer-feasible solution to the problem. We note that this method does not solve the original model fully to optimality, except in the rare case that the optimal solution to the LP is integer feasible. This would require the use of a branch-and-price algorithm (Barnhart *et al*, 1998). However, solving the linear relaxation to optimality does yield a lower bound with which to compare the best found integer feasible solution to.

Case studies

This project was motivated by several case studies provided by FPInnovations, a Canadian not-for-profit organization that carries out scientific research and technology transfer for the Canadian forest industry. Four different case studies, three from Quebec and one from British Columbia, with an annual planning horizon were provided, and we will denote these A1 through A4. All four studies have a yearly planning horizon partitioned into 26 periods of 14 days. Two classes of truck are available to transport the wood. The smaller class of truck is equipped with its own crane for loading wood. They are usually only used for cleaning up smaller piles of wood to avoid the need to operate a loader at a forest site, as they are less cost-efficient per cubic metre of wood to operate (with roughly one-third of the carrying capacity) and generally are slower to load. Data sets A2 and A3 have very inconsistent harvest schedules and hence the schedule balancing Constraints (10) and (11) could not be added while still allowing a mathematically feasible model; however, these were included for data sets A1 and A4.

Table 1 Description of case studies

Instances	F	M	L	V
W1	6	5	3	29 745
W2	6	5	3	16 065
A1	43	7	5	722 531
A2	8	1	1	372 670
A3	8	1	2	462 272
A4	3	1	3	743 600

We additionally were provided two weekly instances, which we denote *W1* and *W2*. We partitioned the week into 7 periods of a single day. The company that provided these instances only utilized a single truck class, not equipped with an onboard loader. In these weekly instances, the scheduling Constraints (10) and (11), and the wood freshness Constraint (6), were not relevant to the model.

For each data set, we provide in Table 1 the number of forest sites (*|F|*), mills (*|M|*), log assortments (*|L|*), and the total volume of wood harvested across all forests over the planning horizon (*V*, in m^3). Distances between forests and mills in all data sets ranged from 1 to 6 hours.

Experimental results

As loading and unloading times were 40 min in all cases, the most intuitive discretization step to use when modelling the problem was 40 min; hence, loading and unloading operations took a duration of 1 timestep, and a day had a time dimension of size 36. We implemented the algorithms in C++, and used Gurobi 4.6 as an LP and MIP solver. All Gurobi parameters were set to defaults except, during the column generation phase of the algorithm, we solved the linear programmes using the barrier optimizer to generate interior solutions and hence yield more useful dual values to take to the subproblems. All experimentation was done on an Intel Core i7, 2.67 GHz processor with 4.0 GB of memory.

We solved each of the annual case studies varying the parameters n_p^L that represent the maximum number of sites opened for loading, which we held constant across all periods, from 0 (in which case strictly self-loading trucks could be utilized) up to a maximum based on the size of the data set, after which the number of loaders did not further constrain the instance. In addition, we varied the parameter ϵ to control schedule balancing through parameters ∞ (no balancing), 0.5, and 0.25 on the relevant instances. All runtimes were limited to 60 min, with 30 min devoted to column generation and any remaining time devoted to solving the MIP. In Table 2, we display the total size of the route pool after stopping the column generation, the total number of variables and constraints in the MIP, the objective value of the best feasible solution (if one was determined), the optimality gap (if the LP was solved to optimality), and the computation time.

Table 2 Results on case studies

n_p^L	ϵ	J	Variables	Constraints	Objective	Gap (%)	Time (s)
A1							
0	∞	4891	596 614	533 555	13 625 183	0.61	697
0	0.5	7213	629 122	534 284	13 627 674	0.50	968
0	0.25	8043	640 742	534 284	13 648 664	0.60	951
2	∞	8620	648 840	533 555	9 655 621	6.58	3600
2	0.5	7493	633 042	534 284	—	—	3600
2	0.25	8350	645 040	534 284	—	—	3600
4	∞	7172	628 548	533 555	8 365 557	2.15	3600
4	0.5	7767	636 878	534 284	—	—	3600
4	0.25	8397	645 698	534 284	8 389 367	2.03	3600
6	∞	7724	636 276	533 555	8 303 164	1.40	3600
6	0.5	8152	642 268	534 284	8 317 962	—	3600
6	0.25	7825	637 690	534 284	8 326 780	—	3600
8	∞	7660	635 380	533 555	8 293 451	1.28	3600
8	0.5	8211	643 094	534 284	8 308 366	1.26	3600
8	0.25	9225	657 290	534 284	8 328 978	1.30	3600
A2							
0	∞	2078	205 945	264 865	9 682 963	0.09	33
1	∞	1192	193 541	264 865	4 260 043	1.90	73
2	∞	809	188 179	264 865	4 140 578	0.65	47
A3							
0	∞	1415	207 407	280 870	10 202 636	0.11	32
1	∞	1846	213 441	280 870	6 596 209	6.85	120
2	∞	1102	203 025	280 870	6 017 224	0.50	47
A4							
0	∞	1268	202 806	277 129	7 059 439	0.18	12
0	0.5	1455	205 426	277 858	7 084 142	0.48	33
0	0.25	1301	203 270	277 858	7 072 485	0.27	33
1	∞	1381	204 388	277 129	4 447 259	18.24	3600
1	0.5	2502	220 084	277 858	4 088 901	8.56	2832
1	0.25	2409	218 782	277 858	4 234 101	12.11	3600
2	∞	868	197 206	277 129	3 791 177	0.80	297
2	0.5	1383	204 418	277 858	3 793 560	0.76	472
2	0.25	1325	203 606	277 858	3 823 025	1.45	3600
3	∞	896	197 598	277 129	3 786 397	0.22	81
3	0.5	1344	203 872	277 858	3 795 472	0.68	69
3	0.25	1278	202 948	277 858	3 962 911	4.55	3600

It is clear that the existence of a heterogeneous truck fleet adds an additional level of complexity to this problem. When the number of loaders is sufficiently small or large, generating high-quality solutions becomes a significantly easier task as the subproblems generate nearly exclusively trucks from a single class. When this is not the case, however; solutions took longer to find, and in a few cases a feasible solution could not be found under the imposed time constraint. Looking at case study *A4* specifically, which has the largest volume, the problem is most difficult to solve when one loader is present, and the final gaps are quite large. This enlarged solution space (from requiring both classes of truck) does not allow us to find a good feasible solution under the given time constraints. In addition, more tightly constraining the balanced schedule requirement has an effect on performance.

For the annual case studies, we provide a breakdown of the distribution of the four objective component costs in Figure 1.

We note that as the routing costs are the most significant, it further motivates taking these costs into account during tactical planning. Storage costs are also quite large in magnitude, but with supply and demand both deterministic there is not as much flexibility to minimize these costs.

For the case study A1, storage costs (\$0.5 per cubic metre at the mill and \$0.2 per cubic metre at the forest roadside) accounted for approximately \$2 million, or between roughly 15–25% of the objective value. We provide in Figure 2 the inventory (cumulatively over all log assortments) over time at each mill, and cumulatively at all forest roadides. We note a high variance over the planning horizon, and smoothing this has been identified as a priority for future work.

We then assessed the performance of this methodology (*ColGen*) by comparing it with the aforementioned two-phase flow-based approach (*Flow*) (El Hachemi, 2011). In the flow-based approach, it was necessary to generalize the methodology slightly in the case of annual plans to account for periods of multiple days by adding an additional index to the flow variables. In order to apply this methodology, the model also

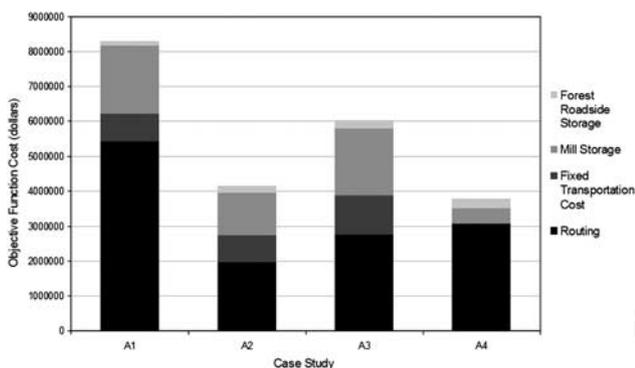


Figure 1 Objective function component costs per case study.

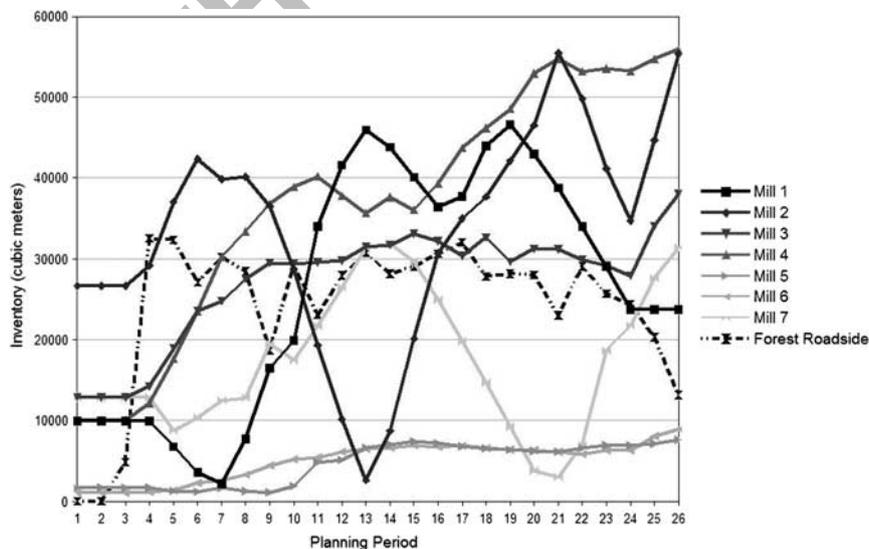


Figure 2 Inventory at mills and forest roadside in an industrial problem.

needed to be slightly adjusted. We assumed a homogeneous fleet of trucks, not equipped with onboard loaders, and hence made the conversion of all volumes to truckloads. In addition, it is not possible to apply the schedule balancing constraints to a decomposed approach, so those constraints were not included for this comparison.

For both methodologies, we limited the runtime to 40 min on the annual sets and 20 min on the weekly sets. In all cases, the column generation was able to solve the linear relaxation to optimality, and find a near-optimal integer solution. We provide in Table 3 the optimality gap of this solution, and additionally the improvement of this solution over that provided by the decomposed methodology.

In five of six cases, the column generation methodology finds an improved solution. Although the improvement is not very significant in several cases, we are able to solve a much more robust problem, and are able to find solutions in examples where a decomposition fails. To illustrate, we consider the instance A4. When solving the tactical phase of the model, one must bound the number of pickups that can be made from a forest and deliveries that can be made to a mill because of the limited number of hours a loader can be operational. If this expression is constrained too tightly, then the tactical model has the potential to be infeasible. However, if it is not constrained tightly enough, then the resulting tactical solution may yield an infeasible operational subproblem. The existence of different operating hours at different locations only increases the difficulty of these decisions.

Implementation into decision support system

This project was undertaken in collaboration with FPIInnovations to develop a complementary tool to the current FPSuite software package. The FPIInterface module of the FPSuite is an

Table 3 Comparison of methodologies

Instances	Gap (%)	Colgen improvement(%)
W1	0.87	1.05
W2	3.53	-2.42
A1	0.11	0.55
A2	0.28	8.25
A3	0.12	3.43
A4	0.17	—

operational tool used to model the forest supply chain to estimate costs for road construction and maintenance, harvesting, transport, and reforestation.

Therefore, the goal was to create a decision support system that can be used to formulate a complete plan in a short computational time, with the option to easily modify inputs to perform sensitivity analysis on transportation costs under many different scenarios. Detailed reports must be given in the form of an Excel workbook, allowing the user to track inventories, allocations, and driver and loader schedules throughout the planning horizon, and to easily calculate important key performance indicators such as the average utilization hours of each truck and the ratio of loaded to unloaded travel time. The software developed in this project can be utilized in a stand-alone fashion, or take as inputs the output provided by the FPInterface, to produce these required schedules.

Being able to generate these efficiently and in a simple interface has allowed FPInnovations researchers to independently use this tool in practical settings in order to run scenarios for many Canadian forestry companies. FPInnovations is currently negotiating with several of these companies to pursue the implementation of this optimization tool in their operations.

Conclusion and future work

We have introduced a multi-period tactical wood flow model in which routing and scheduling decisions are incorporated to maximize savings generated from backhaul opportunities and queuing minimization. This problem was modelled as a mixed integer linear programme and we solved the linear relaxations via column generation, using dynamic programming to solve the subproblems at each master iteration. Integer feasible solutions were then found via branch-and-bound after solving the linear relaxations to optimality. On problem sets provided by an industrial partner, very good integer feasible solutions were found within a reasonable time limit. The methodology also outperforms a previously developed decomposed approach. We believe that not decomposing the model is justified in many cases, and that column generation is a powerful tool to generate truck schedules.

Future work involves synchronizing the routing decisions with other aspects of the supply chain, such as harvest planning.

In addition, driver-specific constraints such as shift length and geographical restrictions have been identified as a priority for future research efforts.

Acknowledgements—This work falls in line with Theme 4 of the NSERC Strategic Research Network on Value Chain Optimization (VCO). The authors would like to thank the VCO Network for their collaborative efforts, and Mitacs and the FQRNT for their financial support. In addition, we would like to thank Dave Lepage from FPInnovations for his contributions in development and testing.

References

- Audy J-F, Amours SD and Ronnqvist M (2012). Planning methods and decision support systems in vehicle routing problem for timber transportation: A review. Technical Report CIRRELT-2012-38, CIRRELT, 2012.
- Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MWP and Vance PH (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research* **46**(3): 316–329.
- Berbeglia G, Cordeau J-F, Gribkovskaia I and Laporte G (2007). Static pickup and delivery problems: A classification scheme and survey. *TOP* **15**(1): 1–31.
- Coelho LC, Cordeau J-F and Laporte G (2012). Thirty years of inventory-routing. Technical Report CIRRELT-2012-52, CIRRELT.
- Cordeau J-F, Laporte G and Mercier A (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society* **52**(8): 928–936.
- Cormen T, Leiserson C, Rivest R and Stein C (1990). *Introduction to Algorithms*. The MIT Press: Cambridge, MA.
- Dror M, Laporte G and Trudeau P (1994). Vehicle routing with split deliveries. *Discrete Applied Mathematics* **50**(3): 239–254.
- El Hachemi N (2011). Problème de transport avec contraintes horaires. PhD thesis, École Polytechnique de Montréal, December.
- El Hachemi N, Gendreau M and Rousseau L-M (2013). A heuristic to solve the synchronized log-truck scheduling problem. *Computers and Operations Research* **40**(3): 666–673.
- Epstein R, Rönnqvist M and Weintraub A (2007). Forest transportation, chapter 20. *Handbook of Operations Research in Natural Resources*. Springer, pp 391–403.
- FPSuite (n.d.). An integrated process control platform for forestry operations. <http://www.fpsuite.ca>, accessed November 2013.
- Gémieux G (2009). Planification de la récolte et allocation des produits aux usines. Masters thesis, Université de Montréal, August.
- Gingras C, Cordeau J-F and Laporte G (2007). Un algorithme de minimisation du transport à vide appliqué à l'industrie forestière. *Information Systems and Operational Research* **45**(1): 41–47.
- Gronalt M and Hirsch P (2007). Log-truck scheduling with a tabu search strategy. In: Doerner KF, Gendreau M, Greistorfer P, Gutjahr WJ, Hartl RF and Reimann M (eds) *Metaheuristics, Volume 39 of Operations Research/Computer Science Interfaces Series*. Springer: New York, pp 65–88.
- Gunnarsson H, Rönnqvist M and Carlsson D (2007). Integrated production and distribution planning for Södra Cell AB. *Journal of Mathematical Modelling and Algorithms* **6**(1): 25–45.
- Karlsson J, Rönnqvist M and Bergström J (2004). An optimization model for annual harvest planning. *Canadian Journal of Forest Research* **34**(8): 1747–1754.
- Linnainmaa S, Savola J and Jokinen O (1995). EPO: A knowledge based system for wood procurement management. In *Proceedings of the*

- seventh Annualconference on Artificial Intelligence*, Vol. 5015, Montréal, pp 107–113.
- Natural Resources Canada (2011). Ressources Naturelles Canada, <http://www.nrcan-mcan.gc.ca>, accessed November 2013.
- Palmgren M, Rönnqvist M and Värbrand P (2004). A near-exact method for solving the log-truck scheduling problem. *International Transactions in Operational Research* **11**(4): 447–464.
- Rey PA, Muñoz JA and Weintraub A (2009). A column generation model for truck routing in the Chilean forest industry. *Information Systems and Operational Research* **47**(3): 215–221.
- Rix G, Rousseau L-M and Pesant G (2011). Solving a multi-period log-truck scheduling problem with column generation. Presented at the 34th Council on Forest Engineering Annual Meeting. Quebec, QC, Canada.
- Rönnqvist M (2003). Optimization in forestry. *Mathematical Programming* **97**(1–2): 267–284.

Appendix

Shortest Path Dynamic Programming Algorithm

Lines 1 through 6 initialize the labels. Lines 7 through 11 push through the graph and update labels as required. Lines 12

through 15 store all negative reduced cost routes originating and terminating at mill m .

Algorithm 1 Shortest Dynamic Programming Path Algorithm

```

1: for all  $v$  in  $V$  do
2:    $pred_v \leftarrow \text{null}$ 
3:   if  $v$  corresponds to mill  $m$  then
4:      $RC_v \leftarrow \lambda_{\text{mpdt}} - \gamma_{\text{pd}} + \gamma_{\text{p,d}+1}$ 
5:   else
6:      $RC_v \leftarrow \infty$ 
7:   for all  $u$  in  $V$  following the topological ordering do
8:     for all  $(u, v)$  in  $A$  do
9:       if  $RC_v < RC_u + c_{uv}$  then
10:         $RC_v \leftarrow RC_u + c_{uv}$ 
11:         $pred_v \leftarrow u$ 
12:   for all  $v$  in  $V$  do
13:     if  $v$  corresponds to mill  $m$  then
14:       if  $RC_v < 0$  then
15:         Iterate backwards and store path to  $v$ .
```

Received 15 May 2012;
accepted 18 November 2013 after two revisions

AUTHOR COPY