

PROGRAMACIÓN SIMULTÁNEA DE BUSES Y CONDUCTORES:
CASO DE ESTUDIO EN TRANSANTIAGO, CHILE

Simultaneous vehicle and crew scheduling problem: case study Transantiago, Chile

César Augusto Henao

Doctor en Ciencias de la Ingeniería. Pontificia Universidad Católica de Chile, Departamento de Ingeniería de Transporte y Logística. Universidad del Norte, Departamento de Ingeniería Industrial, Colombia.
cahenao@uc.cl

Rodolfo Alejandro Cuevas

Magíster en Ciencias de la Ingeniería. SHIFT: Workforce Management, Chile. rcuevas@shiftlabor.com

Cómo citar / How to cite

Henao, C. y Cuevas, R. (2016). Programación simultánea de buses y conductores: caso de estudio en Transantiago, Chile. *Revista CEA*, 2(4), 11-25.

Recibido: 28 de febrero de 2016

Aceptado: 15 de mayo de 2016

Resumen

En este artículo proponemos una formulación alternativa para resolver el problema de programación simultánea de buses y conductores. Este trabajo presenta una heurística constructiva y un modelo de programación lineal entera mixta para abordar este problema. Para disminuir sustancialmente los tiempos de solución, nuestra formulación construye de manera implícita el conjunto óptimo de itinerarios de buses. Adicionalmente, el modelo incorpora componentes de costo en la función objetivo, que permiten mejorar la calidad operacional de la solución entregada. Nosotros presentamos los resultados de la implementación de nuestra metodología para un caso de estudio que usa instancias reales de uno de los principales operadores privados de buses en Transantiago, Chile. Los resultados del modelo evalúan el *trade-off* entre las dos soluciones extremas del problema abordado: priorizar la minimización de turnos *versus* priorizar la minimización de itinerarios.

Palabras claves: transporte público; programación de vehículos y turnos; programación lineal entera mixta.

Abstract

In this paper we propose an alternative formulation to solve the simultaneous vehicle and crew scheduling problem. This paper presents a constructive heuristic and a mixed integer linear programming model to address this problem. To substantially reduce solution times, our formulation proposes a way to implicitly construct the optimal set of vehicle schedules. Additionally, the objective function of the model incorporates multiple cost terms that improve the operational quality of the delivered solution. We present the results of the implementation of our methodology for a study case using real instances from one of major private bus operators in Transantiago, Chile. The model results evaluate the trade-off between two extreme solutions from addressed problem: prioritizing minimization of shifts versus prioritizing minimization of vehicle schedules.

Keywords: public transport; Vehicle and crew scheduling; Mixed integer linear programming.

1. INTRODUCCIÓN

Los problemas de programación de vehículos (VSP, por sus siglas en inglés) y asignación diaria de turnos a conductores (CSP, por sus siglas en inglés) son importantes problemas de optimización combinatorial que surgen en el proceso de planificación de operadores de transporte público (Mesquita y Paias, 2008). Los operadores deben satisfacer una demanda diaria de viajes para cada línea de buses que operan y a la vez coordinar eficientemente sus recursos principales: buses y conductores. Bunte y Kliewer (2009) explican que ambos problemas son considerados de complejidad NP-hard en el sentido combinatorial, y han sido un tópico de intenso estudio en el área de investigación operacional. Tradicionalmente debido a su naturaleza compleja, estos problemas son resueltos de forma separada o secuencial (Lauren y Hao, 2008; Mesquita y Paias, 2008).

Muchos autores también han abordado el problema de resolver simultáneamente el VSP y el CSP, denominándolo VCSP (e.g., Gaffi y Nonato, 1999; Huisman et al., 2005; Mesquita y Paias, 2008; Kliewer et al., 2012). Sin embargo, aún hay mejoras que pueden ser realizadas en la metodología de solución del VCSP simultáneo. Primero, a excepción de Groot y Huisman (2008), la mayoría de los trabajos han resuelto el VCSP simultáneo para instancias pequeñas o medianas. Segundo, una revisión de los trabajos previos sobre el VCSP simultáneo, muestra que típicamente la función objetivo del sub-problema CSP no incluye componentes de costos de importante relevancia práctica. Su inclusión en la función objetivo permitiría mejorar la calidad de los turnos de conducción e itinerarios que pertenecen a la solución. Los turnos son tareas asociadas a los conductores y los itinerarios son tareas asociadas a los buses. Tal que un itinerario puede estar conformado por uno o más turnos.

La contribución de este trabajo consiste en desarrollar un enfoque de solución novedoso que permita resolver el VCSP de forma simultánea para un caso de estudio real. La

metodología de solución propuesta está compuesta por dos etapas y permite obtener soluciones de buena calidad y resolver instancias reales en tiempos razonables. En una primera etapa, se usa una heurística constructiva para generar un conjunto de turnos candidatos a ser parte de la solución por día tipo (i.e. días laborales, sábado, domingo) y cada línea de buses de la red de transporte público que cubre el operador. En la segunda etapa, un Modelo de Programación Lineal Entera Mixta (MPLEM) resuelve el VCSP simultáneo por día tipo y línea de buses, en un horizonte de planificación semanal. El MPLEM toma como input el conjunto de turnos candidatos generados en la primera etapa y posteriormente selecciona la combinación óptima de turnos e itinerarios que satisface la demanda diaria de viajes comerciales para cada línea de buses.

La segunda etapa es particularmente interesante por tres razones. Primero, a diferencia de otros autores (e.g. Hasse et al., 2001; Huisman et al., 2005; Mesquita y Paias; 2008), nosotros usamos una formulación más simple que construye de manera implícita el conjunto de itinerarios óptimo, esto permite reducir los tiempos de solución. Segundo, dado que en nuestro caso de estudio los terminales y cabezales están ubicados en el mismo sitio o muy cerca, no es necesario modelar los viajes no comerciales (i.e., viajes sin pasajeros) entre cabezales y terminales, lo cual disminuye el tamaño del problema. Los cabezales son locaciones físicas donde inician o finalizan los viajes, y los terminales son instalaciones donde se guardan los buses. Tercero, el sub-problema CSP incluye componentes de costo en la función objetivo que entregan turnos de mejor calidad. El MPLEM es resuelto mediante un software comercial y sin necesidad de usar técnicas sofisticadas de descomposición como generación de columnas o metaheurísticas.

Finalmente, la metodología propuesta es implementada para un caso de estudio, para ello se usan instancias reales de uno de los principales operadores privados de buses de Transantiago.

Este es el mayor operador central de buses de transporte público urbano en Chile y uno de los principales en Latinoamérica. El caso de estudio consiste en tomar la red de transporte público que cubre el operador y resolver el VCSP por día tipo para cada una de las líneas de buses adjuntas a esta red. El VCSP es resuelto bajo dos experimentos. El primer experimento obtiene la cota máxima de itinerarios cuando el objetivo principal es minimizar los turnos que satisfacen la demanda de viajes. El segundo experimento obtiene la cota mínima de itinerarios cuando el objetivo principal es minimizar la cantidad de buses usados en la operación.

2. REVISIÓN DE LITERATURA

Cualquier proceso de planificación para un operador de transporte público incluye cuatro componentes básicos que usualmente se resuelven secuencialmente (Ceder, 2002). (1) Diseño de líneas, i.e., define un conjunto de líneas o rutas de buses que sirven un área particular, donde cada línea sirve un recorrido y una secuencia de paradas. (2) Construcción del horario maestro de viajes, i.e., establece adecuados horarios de los viajes comerciales para satisfacer la demanda de pasajeros en cada línea, cumpliendo con las restricciones de frecuencias por línea pre-establecidas por el operador central. Cada viaje requiere un vehículo y un conductor. (3) Programación de vehículos (VSP), i.e., conjuntos de viajes son agrupados en una cadena (i.e. itinerario) para ser operados por un solo bus. El objetivo principal es minimizar el número de itinerarios requeridos para satisfacer el horario maestro de viajes. (4) Asignación de conductores, i.e., asigna turnos de trabajo a conductores para satisfacer el horario maestro de viajes. El problema de asignación de conductores es usualmente dividido en dos: (a) asignación de conductores (CSP), el cual asigna una secuencia de viajes diaria para ser trabajada por un solo conductor (i.e. turnos de conducción); y (b) proceso de *rostering*, i.e., una vez los requerimientos diarios de turnos han sido determinados, se realiza una asignación semanal

de estos turnos a conductores específicos (i.e. asignación semanal de turnos de trabajo y días de descanso).

Guihaire y Hao (2008) explican que en el primer componente del proceso de planificación se toman decisiones del tipo estratégico, mientras que en el segundo componente se toman decisiones del tipo táctico, y en los componentes tres y cuatro se toman decisiones a nivel operacional. La solución del VCSP asegura que los vehículos y conductores (i.e. recursos) cubran los viajes programados en el horario maestro de viajes. Bajo un enfoque simultáneo, el VSP y el CSP se resuelven de forma integrada en el mismo modelo. Dado un conjunto de viajes dentro de un horizonte de planificación fijo, el problema consiste en determinar para cada día y línea de buses el programa de mínimo costo para los vehículos y la tripulación de conductores, tal que tanto el VSP como el CSP sean factibles y mutuamente compatibles (Freling et al., 2003). La dificultad del problema es alta porque el tamaño del problema es mucho mayor.

Estudios previos que han tratado la solución simultánea del VCSP son listados en la Tabla 1 junto con breves indicaciones de sus características claves.

1. *Cantidad de Terminales (CT)*. Indica si el problema considera una única terminal (UT) o múltiples terminales (MT) donde puede ser guardado un bus cualquiera.
2. *Flota Heterogénea (FH)*. Indica si el problema considera que hay buses con características distintas (e.g., capacidad de pasajeros, velocidad) o si todos los buses son idénticos.
3. *Función Objetivo VSP (FO-VSP)*. Componentes en la función objetivo que corresponden al VSP.
4. *Función Objetivo CSP (FO-CSP)*. Componentes en la función objetivo que corresponden al CSP.
5. *Método*. Indica que tipo de solución metodológica fue usada.
6. *Enfoques de solución (ES)*. Indica los tipos de enfoques de solución usados para resolver

el VCSP: (i) Independiente (I); (ii) Secuencial (SE); y (iii) Simultáneo (S).

7. *Tamaño de la Instancia* (TI). Indica la cantidad de viajes que contiene la instancia de prueba más grande del estudio.

Tabla 1. Características Seleccionadas de estudios previos sobre el VCSP simultáneo
Table 1. Selected Characteristics of Previous Studies on Simultaneous VCSP

Referencia	CT	FH	FO-VSP	FO-CSP	Método	ES	TI
Gaffi & Nonato (1999)	MT	-	-	-	Relajación lagrangiana, Generación de columnas	SE, S	257
Haase & Friberg (1999)	UT	-	Min. Número de vehículos	Min. Número de turnos	Modelos <i>set partitioning</i> , Algoritmo <i>Branch-&Cut-&Price</i>	S	20
Haase et al. (2001)	UT	No	-	-	Formulación de flujo en redes <i>multi-commodity</i> , Algoritmo <i>Branch & Price</i>	S	350
Freling et al. (2003)	UT	No	Min. Número de vehículos + Tiempo Ocioso + Tempo de viaje	Min. Número de turnos	Programación lineal entera, Formulación <i>quasi-assignment</i> , Modelo <i>set partitioning</i> , Heurísticas lagrangianas, Generación de columnas	I, SE, S	238
Huisman et al. (2005)	MT	No	Min. Número de vehículos + Tiempo Ocioso + Tempo de viaje	Min. Número de turnos	Programación lineal entera, Relajación lagrangiana, Generación de columnas	SE,S	653
Rodrigues et al. (2006)	UT	No	Min. Número de vehículos	Min. Número de turnos + Horas extras	Modelos de programación lineal, Heurísticas	S	-
Groot & Huisman (2008)	MT	No	Min. Número de vehículos + Tiempo Ocioso + Tempo de viaje	Min. Número de turnos	Programación lineal entera, Relajación lagrangiana, Generación de columnas, Algoritmos para dividir grandes instancias	SE, S	1372
Laurent & Hao (2008)	UT	Sí	Min. Número de vehículos	Min. Número de turnos	<i>Constraint-based model</i> , <i>Greedy randomized adaptive search procedure</i>	SE, S	249
Mesquita & Paias (2008)	MT	No	Min. Costo de viajes no comerciales + Número de vehículos	Min. Número de turnos	Formulación de flujo en redes <i>multi-commodity</i> , Modelo <i>set partitioning/covering</i> , Generación de columnas implícita	S	400
Kliewer et al. (2012)	MT	No	Min. Número de vehículos + Tiempo de espera	Min. Número de turnos	Formulación de flujo en redes <i>multi-commodity</i> , Modelo <i>set partitioning</i> , Relajación lagrangiana, Generación de columnas	S	661

Fuente: elaboración de los autores.

La extensa literatura sobre el VCSP simultáneo testimonia el considerable esfuerzo del que ha sido devoto el problema y la relevancia práctica del mismo. Sin embargo, como se mencionó tempranamente en la introducción, aún hay mejoras que pueden ser realizadas en la metodología de solución del VCSP simultáneo. Primero, a excepción de Groot y Huisman (2008),

la mayoría de los trabajos han resuelto el VCSP simultáneo para instancias pequeñas o medianas. Segundo, los estudios previos omiten, especialmente en la función objetivo del sub-problema CSP, componentes de costo que son de importante relevancia a nivel práctico y cuya inclusión permitiría mejorar la calidad de los turnos e itinerarios que pertenecen a la solución

del VCSP. Estos componentes de costos son, por ejemplo: horas hombre, horas extras, horas subvencionadas, penalidad a turnos con salidas nocturnas, penalidad a turnos con muy bajo o altos tiempos de descanso entre viajes, penalidad por déficit de buses.

Finalmente, este estudio propone una metodología a dos etapas para resolver el VCSP simultáneo. El estudio se enmarca dentro del análisis de un caso de estudio que usa instancias pertenecientes a un sistema real. La metodología propuesta permite resolver estas instancias en tiempos razonables, y a su vez, permite incluir componentes de costo en el sub-problema CSP que entregan soluciones de calidad y que se ajustan a las necesidades prácticas de un operador de transporte público.

3. DEFINICIÓN DEL PROBLEMA

El problema que nosotros proponemos estudiar consiste en desarrollar una solución simultánea, rápida, y eficiente para un problema ampliamente desarrollado en la literatura: programación de buses y asignación diaria de turnos a conductores. Para un horizonte de planificación semanal, el problema consiste en determinar la combinación óptima de itinerarios y turnos que satisfaga la demanda diaria de viajes por día tipo y línea de buses para una empresa operadora de transporte público urbana. La solución del VCSP debe garantizar la coherencia espacio-temporal de los viajes que cubren los itinerarios y turnos; y también respetar un conjunto de restricciones impuestas por la ley laboral, acuerdos tácitos de la empresa operadora, y condiciones operacionales del sistema. Nuestra formulación del problema refleja el análisis de un caso de estudio para uno de los principales operadores privados de buses de Transantiago, Chile.

Específicamente, el problema incorpora los siguientes 14 supuestos. 1) El diseño de la red de líneas es conocido e inalterable. 2) El horario maestro de viajes es conocido, i.e., la demanda

de viajes por día tipo y línea de buses. Un viaje es el trabajo mínimo de un conductor, está definido por una hora de inicio y fin, y una terminal de origen y destino. 3) La operación del transporte público se distribuye en varias terminales ubicadas en distintos puntos de la ciudad. Tal que cada conductor y bus pueden pertenecer a una sola terminal. 4) La ubicación de los terminales coincide con los cabezales. Esta característica de la red evita la necesidad de modelar los viajes no comerciales entre cabezales y terminales. 5) La red de transporte público es dividida en varias subredes. Cada subred contiene un subconjunto de terminales de la red y sobre cada subred operan cierta cantidad de líneas diferentes que comparten tales terminales. 6) El problema es separable por líneas buses. Por lo tanto, cada turno e itinerario opera viajes de una sola línea. 7) El terminal de inicio y término para cada turno e itinerario debe ser el mismo. 8) Los turnos e itinerarios están compuestos por dos tipos de tareas: viajes comerciales y viajes no comerciales. La literatura menciona que principalmente hay dos tipos de viajes no comerciales: (i) movimientos de buses entre terminales y cabezales. Este tipo de viajes no es considerado en nuestra formulación, tal como se describió en el supuesto 4). (ii) Movimientos de buses sin pasajeros entre cabezales, nosotros lo denominaremos viaje dummy. Nuestra formulación sí considera la posibilidad de realizar viajes dummy. 9) El VCSP no incorpora en su solución el problema de *rostering* (supuesto usual en la literatura), el cual puede ser resuelto una vez se tiene nuestra solución del VCSP. 10) Inicialmente hay una flota de buses conocida por terminal y línea de buses. 11) La flota de buses se asume homogénea. 12) De acuerdo con los contratos de trabajo definidos, los turnos pueden tener diferentes duraciones a través de los días de la semana, pero no pueden exceder una jornada máxima por día. El concepto de jornada se refiere a la duración del trabajo diario de un conductor.

En nuestra formulación nosotros consideramos dos tipos de jornadas: (i) jornada base, no incluye las horas extras máximas permitidas por día; y (ii)

jornada máxima, sí incluye las horas extras. 13) Cada turno debe respetar una máxima conducción continua permitida, un tiempo mínimo de descanso entre viajes, y un número mínimo y máximo de viajes. 14) Utilizamos el concepto *partes del día* para dividir cada día tipo en ventanas de tiempo disjuntas (e.g., mañana, tarde, noche). Esto implica que los turnos e itinerarios solo pueden iniciar dentro de las partes del día predefinidas para cada terminal.

4. METODOLOGÍA DE SOLUCIÓN

Generación de turnos candidatos

Corresponde a la Etapa 1 de la metodología propuesta en este trabajo. En esta etapa un algoritmo general (ver Algoritmo 1) es ejecutado para cada subred y día tipo. Para cada día tipo, la primera fase del Algoritmo 1 carga información detallada de la subred y de las terminales asociadas a dicha subred. Posteriormente, para cada día tipo y línea de buses de la subred, la segunda fase del Algoritmo 1 invoca y ejecuta la heurística constructiva propuesta para construir el conjunto de turnos candidatos y factibles (ver Algoritmo 2). El conjunto de turnos generados será posteriormente usado como un input en la Etapa 2 de la metodología.

De forma iterativa, la heurística descrita en el Algoritmo 2 crea un conjunto de turnos candidatos a través de la creación de bloques de viajes y su posterior combinación. Los bloques son agrupaciones de viajes que deben ser de ida y vuelta, por lo cual cada bloque está conformado por un número par de viajes. La heurística desarrollada se preocupa por satisfacer dos lógicas principales. Primero, los turnos construidos deben respetar la coherencia espacio-temporal de los viajes que cubren. Es decir, cada par de viajes consecutivos v_1 y v_2 pertenecientes a un turno deben cumplir dos condiciones: (1) el terminal destino del viaje v_1 debe ser el mismo que el terminal origen del viaje v_2 (i.e. coherencia espacial) y (2) el tiempo de término del viaje v_1 debe ser menor que el tiempo de inicio del viaje v_2 (i.e. coherencia temporal). Segundo, se garantiza que cada turno construido respete un conjunto de restricciones impuestas por la ley laboral y acuerdos tácitos del operador, tales como: (1) no exceder la jornada máxima por día; (2) no exceder la máxima conducción continua permitida; y (3) garantizar un tiempo mínimo de descanso entre viajes, incluyendo el periodo de almuerzo.

Algoritmo 1. Procedimiento general de generación de turnos candidatos para cada subred

Fase 1: Inicialización

Paso 1: obtener información de la subred: (i) cantidad y listado de arcos asociados a los viajes; (ii) cantidad y listado de nodos (i.e., terminales o cabezales), (iii) consolidado de líneas de buses; y (iv) demanda viajes por línea de buses.

Paso 2: obtener consolidado de las partes del día definidas para cada terminal y día tipo.

Fase 2: Generación de turnos candidatos por día tipo y línea de buses

Paso 3: proceso iterativo por día tipo que toma una línea de buses del listado total y resuelve la heurística constructiva descrita en el Algoritmo 2.

Paso 4: obtener el conjunto total de turnos candidatos D por día tipo y línea de buses, generado a través de la aplicación del Algoritmo 2.

Algoritmo 2. Procedimiento de generación de turnos candidatos por día tipo y línea de buses

Fase 1: Inicialización

Paso 1: obtener información sobre: (i) Fase 1 del Algoritmo 1; (ii) mínima y máxima cantidad permitida de viajes por bloque; (iii) holgura de tiempo mínima entre viajes consecutivos; (iv) máxima conducción continua de un conductor; (v) jornadas de trabajo factibles según los contratos de los conductores.

Fase 2: Generación de bloques

Paso 2: crear una lista con el total de los viajes a cubrir y una lista donde guardar los bloques construidos.

Paso 3: para un viaje v_1 de la lista:

- a. Se crea un bloque b al cual se le agrega el viaje v_1 , y ese viaje se retira del listado total de viajes pues fue usado.
- b. Para ese viaje v_1 se encuentra un viaje v_2 que pueda ser agregado al bloque b . El viaje v_2 puede ser agregado al bloque b si cumple 4 condiciones: (i) el terminal destino del viaje v_1 debe ser el mismo que el terminal origen del viaje v_2 . (ii) el tiempo de término del viaje v_1 debe ser menor que el tiempo de inicio del viaje v_2 . (iii) la duración total del bloque no debe exceder el parámetro de máxima conducción continua y la jornada máxima de trabajo. (iv) el inicio del viaje v_2 respeta la holgura de tiempo mínima entre viajes consecutivos.
- c. Si el viaje v_2 es válido se agrega a b y a su vez se retira del listado de viajes.

Paso 4: si el parámetro de número de viajes máximo por bloque es mayor a dos, se repiten los pasos 3b, y 3c para el viaje v_2 . El objetivo es buscar nuevos viajes que puedan ser agregados al bloque b , pero asegurando que el número de viajes del bloque sea un número par y no supere el máximo permitido.

Paso 5: los pasos 3 y 4 se repiten hasta que se recorra todo el listado de viajes creado en el paso 2. Finalmente se obtiene un listado de bloques construidos.

Fase 3: Generación de turnos candidatos

Paso 6: crear una lista donde guardar los turnos factibles generados.

Paso 7: por enumeración, para cada bloque b del listado de bloques construidos, se generan todos los turnos factibles a través de la combinación con otros bloques del listado. Un turno será factible si cumple las siguientes condiciones: (i) reglas descritas en el Paso 3b. (ii) garantiza una ventana de tiempo donde el conductor puede almorzar. (iii) el turno inicia y finaliza el mismo día. (iv) el inicio del turno está contenido en alguna de las partes del día habilitadas para la terminal.

Paso 8: obtener el conjunto total de turnos candidatos D para el día tipo y línea de buses evaluada.

Formulación del modelo de optimización

Corresponde a la Etapa 2 de la metodología propuesta. En esta sección nosotros formulamos el MPLEM para resolver el VCSP simultáneo por día tipo y línea de buses. A continuación, presentamos la notación matemática usada:

Conjuntos:

- J : Jornadas base, indexado por j .
- V : Viajes comerciales, indexado por v .
- VD : Viajes dummy, indexado por v , $VD \subseteq V$.
- D : Turnos, indexado por d . Listado de turnos candidatos obtenidos en la Etapa 1.

Conjuntos derivados:

- D_j : Turnos con duración de jornada j , indexado por d , $D_j \subseteq D$.
- D_v : Turnos que cubren el viaje v , indexado por d , $D_v \subseteq D$.
- V_d : Viajes cubiertos por el turno d , indexado por v , $V_d \subseteq V$.
- $V_{v_1}^c$: Viajes compatibles desde el viaje v_1 , indexado por v , $V_{v_1}^c \subseteq V$. Conjunto de viajes con inicio posterior al término del viaje v_1 , tal que el destino del viaje v_1 coincide con el origen del viaje a conectar
- $V_{v_2}^h$: Viajes compatibles hacia el viaje v_2 , indexado por v , $V_{v_2}^h \subseteq V$. Conjunto de viajes con término previo al inicio del viaje v_2 , tal que el origen del viaje v_2 coincide con el destino del viaje que lo conecta
- D_{v_1, v_2} : Turnos que cubren el par de viajes compatibles y consecutivos v_1 y v_2 , indexado por d , $D_{v_1, v_2} \subseteq D$.

Parámetros:

- l_d : Duración del turno d , $\forall d \in D$
- w_j : Duración de la jornada base j , $\forall j \in J$
- p_d^{st} : Penalidad por salida tardía del turno d , $\forall d \in D$. Se activa para aquellos turnos que finalizan después de las 24 horas
- p_d^{si} : Penalidad por subvención inicial del turno d , $\forall d \in D$. Se activa para aquellos turnos donde el inicio de su primer viaje es mayor a un tiempo setup preestablecido por terminal
- b : Cantidad de buses disponibles asociados a la línea de buses

Parámetros de costos en la función objetivo:

- c^A : Costo de utilización de un turno
- c^T : Costo de una hora hombre
- c^X : Costo de una hora extra
- c^Ω : Costo de una hora subvencionada
- p^Γ : Ponderación para el costo de penalidad por salida tardía
- p^Ψ : Ponderación para el costo de penalidad por subvención inicial
- p^Λ : Costo de penalidad por cobertura de un viaje dummy
- c^I : Costo de un itinerario
- p^Δ : Costo de penalidad por bus extra

Variables:

- x_d : Igual a 1 si utiliza el turno d , de otra forma 0, $\forall d \in D$
- λ_v : Igual a 1 si el viaje dummy v es no cubierto, de otra forma 0, $\forall v \in VD$
- γ_{v_1, v_2} : Igual a 1 si el viaje v_1 se conecta con el viaje v_2 , de otra forma 0, $\forall v_1 \in V, v_2 \in V_{v_1}^c$
- ω_v : Igual a 1 si el último viaje v de un turno se conecta posteriormente al primer viaje de otro turno, 0 si es el viaje final de un itinerario, $\forall v \in V$
- Δ : Contabiliza el déficit de buses. Es decir, muestra la cantidad de itinerarios que están por encima de los buses iniciales disponibles.

Función objetivo

La función objetivo (FO) (1) minimiza los siguientes términos de costos totales diarios: (a) utilización de los turnos; (b) horas hombre; (c) horas extras; (d) horas subvencionadas; (e) penalización por salidas tardías; (f) penalización por subvención inicial; (g) penalización por cobertura de viajes dummy; (h) itinerarios: cantidad de buses que se necesitan para realizar

el plan operacional multiplicado por un costo muy grande por bus requerido. (i) costo de déficit de buses: penaliza fuertemente la generación de itinerarios para los cuales no hay buses físicos. Los términos (a) – (g) corresponden a la FO del CSP y los términos (h) y (i) corresponden a la FO del VSP. Con el fin de simplificar la notación de la FO, nosotros definimos un conjunto de variables auxiliares presentadas a continuación:

$$\text{Min } c^A A + c^T T + c^X X + c^\Omega \Omega + p^\Gamma \Gamma + p^\Psi \Psi + p^\Lambda \Lambda + c^I I + p^\Delta \Delta \quad (1)$$

(a) (b) (c) (d) (e) (f) (g) (h) (i)

$$A = \sum_{d \in D} x_d \quad (2)$$

$$T = \sum_{j \in J} \sum_{d \in D_j} x_d l_d \quad (3)$$

$$X = \sum_{j \in J} \sum_{d \in D_j} x_d \max(0, l_d - w_j^b) \quad (4)$$

$$\Omega = \sum_{j \in J} \sum_{d \in D_j} x_d \max(0, w_j^b - l_d) \quad (5)$$

$$\Gamma = \sum_{d \in D} p_d^{st} x_d \quad (6)$$

$$\Psi = \sum_{d \in D} p_d^{si} x_d \quad (7)$$

$$\Lambda = \sum_{v \in VD} \sum_{d \in D_v} x_d \quad (8)$$

$$I = \sum_{v \in V} (1 - \omega_v) - \sum_{v \in VD} \lambda_v \quad (9)$$

La variable (2) calcula el total de turnos utilizados (i.e., hacen parte de la solución). La variable (3) calcula el total de horas hombre. La variable (4) calcula el total de horas extras. Considera solo los turnos cuya duración supera la duración de la jornada base. La variable (5) calcula el total de horas subvencionadas. Considera solo los turnos cuya duración es menor a la duración de la jornada base. La variable (6) calcula la penalidad total por salidas tardías. La variable (7) calcula la penalidad total por subvención inicial. La variable (8) calcula el total de turnos utilizados para la cobertura de viajes dummy. La variable (9) calcula el total de itinerarios que se necesita para cumplir con el horario maestro de viajes. Como

está penalizada fuertemente en la FO se busca que sea el menor valor posible, para ello el modelo tratará de que ω_v sea igual a 1 para muchos viajes (i.e. conectar turnos), pues de esta forma el primer término en (9) será más pequeño. El segundo término corrige la cantidad total de itinerarios, al descartar los viajes dummy no cubiertos ($\lambda_v = 1$). Es decir, cada viaje dummy v siempre corresponde al último viaje de un turno, por lo tanto si $\omega_v = 0$, el viaje dummy v es no conectado, pero esto sólo se traducirá en un itinerario si el viaje dummy fue cubierto por algún turno ($\lambda_v = 0$).

La estructura de costos presentada en este trabajo se diferencia a las usadas en trabajos previos (ver Tabla 1). Nuestra FO incluye términos de costos típicamente no usados, pero que pueden tener una importante relevancia a nivel operacional, tales como: (b), (c), (d), (e), (f), (g), e (i). En especial, los términos (b) – (g) de la FO del CSP son un criterio de selección secundario para obtener el conjunto de turnos óptimo. Es decir, el término (a) domina la FO del CSP, sin embargo, ante la presencia de múltiples soluciones, los términos (b) – (g) funcionan como un criterio de desempate que orienta a la consecución de turnos de mejor calidad.

Restricciones

- *Restricciones estructurales del CSP*

Nuestra formulación se basa en un problema *set partitioning*, y por lo tanto garantiza que cada viaje comercial v sea cubierto por exactamente un turno (i.e. restricciones (10)). Por otra parte, las restricciones (11) dicen que si el viaje dummy v es no cubierto ($\lambda_v = 1$) entonces ningún turno d de la solución incluirá tal viaje.

$$\sum_{d \in D_v} x_d = 1 \quad \forall v \in \{V - VD\} \quad (10)$$

$$\sum_{d \in D_v} x_d = 1 - \lambda_v \quad \forall v \in VD \quad (11)$$

- *Restricciones estructurales del VSP*

Las restricciones (12) dicen que cualquier viaje puede ser conectado o no con un viaje posterior. Las restricciones (13) dicen que cualquier viaje puede ser conectado o no por un viaje previo.

$$\sum_{v_2 \in I_{v_1}^c} \gamma_{v_1, v_2} \leq 1 \quad \forall v_1 \in V \quad (12)$$

$$\sum_{v_1 \in I_{v_2}^h} \gamma_{v_1, v_2} \leq 1 \quad \forall v_2 \in V \quad (13)$$

Las restricciones (14) garantizan que, si un turno d es utilizado, cada viaje del turno será conectado con aquel viaje posterior con tiempo

de inicio más cercano, a excepción del último viaje del turno, que podría ser conectado o no con el viaje inicial de algún otro turno posterior. Notar que se requiere una restricción para cada par de viajes compatibles y consecutivos v_1 y v_2 que hagan parte de algún turno candidato.

$$\gamma_{v_1, v_2} \geq \sum_{d \in D_{v_1, v_2}} x_d \quad (14)$$

$$\forall v_1 \in V, v_2 \in V_{v_1}^c : |D_{v_1, v_2}| > 0$$

Las restricciones (15) dicen que cada viaje v puede conectar a lo sumo un viaje de otro turno. Las restricciones (14) y (15) garantizan implícitamente para cada turno en la solución, que solo el viaje final del turno puede ser conectado con algún viaje inicial compatible de otro turno. Por definición $\omega_{v_1} = 0$ implica el requerimiento de un itinerario. Notar que la construcción de los itinerarios se logra implícitamente solo a través de la conexión de viajes y no a través de la conexión explícita de turnos. Esto es muy importante, pues explorar el conjunto V es mucho más tratable computacionalmente que la exploración del conjunto D . El conjunto D_{v_1, v_2} y las restricciones (14) y (15) permiten diferenciar de manera implícita el último viaje de cada turno d , al igual que el resto de viajes del turno.

$$\sum_{v_2 \in I_{v_1}^c} \gamma_{v_1, v_2} \leq \omega_{v_1} \quad \forall v_1 \in V \quad (15)$$

Las restricciones (16) dicen que si no existe ningún turno d que cubra el viaje dummy v , ese viaje dummy quedará sin cobertura y por lo tanto no será conectado con otro viaje. La restricción (17) calcula la cantidad de buses faltantes requeridos para cumplir los itinerarios.

$$\sum_{v_1 \in I_{v_2}^c} \gamma_{v_1, v_2} \leq \sum_{d \in D_{v_2}} x_d \quad \forall v_2 \in VD \quad (16)$$

$$I - \Delta \leq b \quad (17)$$

- *Restricciones de dominio*

$$x_d \in \{0,1\} \quad \forall d \in D \quad (18)$$

$$\lambda_v \in \{0,1\} \quad \forall v \in VD \quad (19)$$

$$0 \leq \gamma_{v_1, v_2} \leq 1 \quad \forall v_1 \in V, v_2 \in V_1^c \quad (20)$$

$$0 \leq \omega_v \leq 1 \quad \forall v \in V \quad (21)$$

$$\Delta \geq 0 \quad (22)$$

Nuestra formulación del VCSP simultáneo contiene $|D| + |VD| + |V| |V_1^c| + |V| + 1$ variables y $4|V| + |V| |V_1^c| + |VD| + 9$ restricciones. Nuestra formulación entrega principalmente dos ventajas en términos de mejorar la tratabilidad computacional del problema. (1) A diferencia de otros autores (e.g. Hasse et al., 2001; Huisman et al., 2005; Mesquita y Paias; 2008), el modelo construye de manera implícita el conjunto óptimo de itinerarios. (2) Dada la estructura del caso de estudio analizado, el modelo omite el sub-problema de asignación de movimientos de buses desde y hacia el (los) terminal(es) en el VSP. Sin embargo, si se estudiara un problema donde la ubicación de los terminales no coincidiera con la ubicación de los cabezales, este sub-problema podría resolverse posteriormente para asegurar la factibilidad de la solución hallada por el VCSP acá propuesto. Estas ventajas aseguran la obtención de un modelo más simple que tiene menos variables y restricciones en comparación a enfoques previamente presentados y ampliamente citados en la literatura (e.g., Freling et al., 2003; Huisman et al., 2005; Mesquita y Paias; 2008). Notar también que dada la naturaleza de las restricciones no es necesario declarar como binarias las variables del VSP: Δ , γ_{v_1, v_2} y ω_v .

Finalmente, estas dos ventajas permiten obtener soluciones en tiempos razonables para instancias reales del VCSP. Notar que la simplicidad otorgada por la ventaja (2), también puede

denotar que nuestro modelo es incompleto. Por lo tanto, infortunadamente nuestra modelación no puede ser totalmente comparable con trabajos previos que sí incorporan el sub-problema acá omitido. Sin embargo, aun con esta limitación, nuestra formulación implícita (i.e., ventaja (1)) puede ser usada en cualquier problema VCSP, y permite mejorar la tratabilidad computacional del problema por si sola.

5. RESULTADOS Y DISCUSIÓN

Esta sección describe los dos experimentos realizados para evaluar el desempeño de la metodología propuesta: (1) Cota máxima de itinerarios y (2) Cota mínima de itinerarios. Por una parte, la cota máxima de itinerarios se obtiene cuando se asigna un valor mínimo a las componentes de costo de la función objetivo del VSP (e.g., $c^A = \$10000 / \text{turno}$ y $c^I = \$1 / \text{itinerario}$). Por otra parte, la cota mínima de itinerarios se obtiene cuando se pondera mucho más alto el costo de utilización del itinerario en relación al costo de utilización del turno (e.g., $c^A = \$100 / \text{turno}$ y $c^I = \$10000 / \text{itinerario}$).

La metodología descrita previamente fue aplicada a un caso de estudio que usa instancias reales de uno los principales operadores privados de buses de Transantiago, Chile. Los datos y parámetros usados fueron suministrados directamente por dicho operador, y corresponden a la data usada en su operación real. Nuestro enfoque fue implementado en C# y el modelo (1)-(22) fue resuelto, para cada línea de buses y día tipo, usando el software comercial ILOG CPLEX 12.5.1 ejecutándose en un servidor Intel-Xeon E5-4640 con 4 procesadores de 2,2 GHz, 32 núcleos y 192 GB de RAM.

El caso de estudio considera un operador que debe gestionar 54 líneas de buses sobre una red de transporte público que involucra 15 terminales. Bajo nuestro enfoque, esta red de transporte es dividida en 8 subredes. Las líneas de buses en cada subred comparten los mismos

terminales y cada subred opera una cantidad diferente de líneas de buses. Todas las líneas de buses pertenecientes a una misma subred tienen la misma configuración de parámetros en el modelo, y esta configuración está determinada por la línea de buses más restrictiva de la subred.

La Fig.1 muestra de manera conjunta los resultados obtenidos para los dos experimentos evaluados. Estos resultados son presentados para cada combinación de subred y día tipo. Adicionalmente, cada uno de los gráficos presentados en la Fig.1 tiene dos ejes verticales, el primero de ellos mide el *Delta Itinerarios* y el segundo mide el *Delta Turnos*. El *Delta Itinerarios* se calcula como la diferencia entre la cantidad de itinerarios o buses requeridos en la solución del

VCSP para dicha subred ($\sum I$) y la cantidad de buses disponibles en la subred ($\sum b$). El *Delta Turnos* se calcula como la diferencia entre la cantidad de turnos requeridos en la solución del VCSP para el segundo experimento (i.e., cuando se prioriza la minimización los itinerarios) y la cantidad de turnos requeridos en la solución del VCSP para el primer experimento (i.e., cuando se prioriza la minimización los turnos). Cuando los valores de delta son positivos significa que hay un déficit de buses o conductores, mientras que cuando los valores de delta son negativos significa que hay un exceso de estos recursos respecto al valor de referencia.

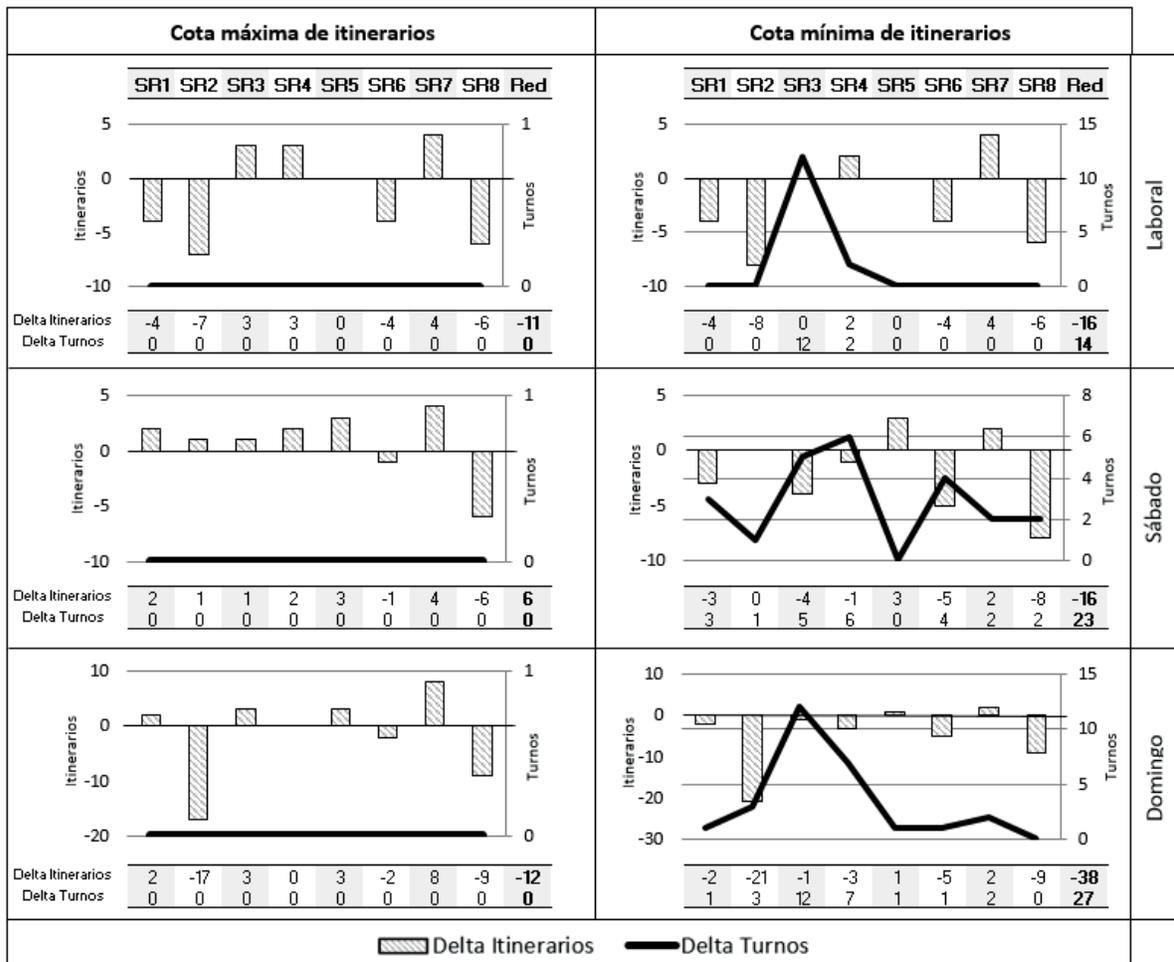


Figura 1. Resultados para Cada Experimento: Itinerarios vs. Turnos
 Figure 1. Results for Each Experiment: Bus Schedules vs. Shifts

Cota máxima de itinerarios

Varios resultados son dignos de destacar en relación con el primer experimento. Primero, es claro que el *Delta Turnos* siempre será igual a cero, ya que esta solución representa la mejor solución posible para el CSP y también representa el punto de comparación respecto de los resultados del segundo experimento.

Segundo, se puede observar que independiente del día tipo, hay disparidad entre los requerimientos de itinerarios de las subredes. Es decir, mientras algunas subredes tienen un déficit de buses otras subredes tienen excesos. Adicionalmente, para el día sábado, la suma del *Delta Itinerarios* para todas las subredes indica que hay un déficit de 6 buses en la red. De hecho, para el día sábado las subredes 1, 2, 3, 4, 5, y 7 tienen déficit de buses.

Cota mínima de itinerarios

También hay resultados dignos de destacar en relación al segundo experimento. Primero, se puede ver que una vez se prioriza la minimización de itinerarios hay un claro aumento en la cantidad de turnos de conducción respecto de los resultados del primer experimento. Es decir, la solución del VSP mejora en detrimento de la solución del CSP. Sin embargo, este déficit se concentra en la operación de las subredes 3 y 4.

Segundo, se puede observar que, en relación con el primer experimento, hay una disminución

significativa en el déficit de buses. Por una parte, en términos agregados (i.e., la red completa) no hay déficits de buses para ningún día tipo. De hecho, es interesante notar que para los días sábado y domingo hay un alto exceso de buses en comparación con el primer experimento. Por otra parte, en términos desagregados, se puede observar que la subred 3 ya no tiene déficit de buses para ningún día tipo.

Tiempos computacionales

La Tabla 2 presenta los tiempos de solución en CPLEX obtenidos para cada experimento y combinación de subred y día tipo. De igual forma, se presentan los resultados agregados para la red completa. Notar que al lado de la etiqueta que distingue cada una de las 8 subredes, también se indica la cantidad de líneas que opera la respectiva subred.

La Tabla 2 muestra claramente que encontrar la solución óptima para el VCSP toma mucho más tiempo cuando se exige en la función objetivo priorizar la minimización de itinerarios (i.e., Experimento 2). De hecho, si se considera la red completa y el consolidado de los tres días tipo, los resultados indican que el tiempo de solución total obtenido en el segundo experimento (i.e., 15.284 seg = 4,24 horas) aumentó un 99,82% respecto al tiempo de solución total obtenido en el primer experimento (i.e., 27,2 seg). Sin embargo, en términos absolutos, nuestro enfoque de solución permite resolver el problema en un tiempo bajo, aun cuando el tamaño y complejidad del problema es grande.

Tabla 2. Tiempos de Solución en CPLEX (segundos): Experimento 1 vs. Experimento 2
Table 2. CPLEX Solution Times (seconds): Experiment 1 vs. Experiment 2

Día	Exp. 1	Exp. 2	Exp. 1	Exp. 2	Exp. 1	Exp. 2	Exp. 1	Exp. 2	Exp. 1	Exp. 2
Tipo	SR1 (7)		SR2 (14)		SR3 (8)		SR4 (7)		Red completa (54)	
Lab	3,0	1991	3,7	616	2,0	1387	0,2	9	Lab	
Sáb	1,1	597	1,5	285	1,1	575	0,1	273	14,9	6548
Dom	0,6	249	1,2	171	1,0	2263	0,1	278	Sáb	
	SR5 (6)		SR6 (4)		SR7 (5)		SR8 (3)		7,3	4661
Lab	0,8	429	2,1	185	3,1	565	0,2	1365	Dom	
Sáb	0,3	583	0,9	1227	1,8	916	0,3	206	5,0	4076
Dom	0,3	15	0,7	274	1,0	813	0,2	12	Total	
									27,2	15284

Fuente: elaboración propia.

6. CONCLUSIONES

Este trabajo presentó a través del análisis de un caso de estudio real, una formulación alternativa para resolver el problema simultáneo de programación de itinerarios de buses y turnos de conducción (VCSP, por sus siglas en inglés). A diferencia de trabajos previos, este artículo propone una formulación más simple para el VCSP que permite construir de manera implícita el conjunto óptimo de itinerarios. Además, dada la estructura del caso de estudio analizado, nuestra formulación no exige modelar los viajes no comerciales desde y hacia el (los) terminal(es).

La formulación propuesta reduce la complejidad del problema combinatorial y permite resolver instancias reales del VCSP simultáneo. Además, nuestra formulación incorpora en la función objetivo del VCSP componentes de costo de importante relevancia a nivel operacional que modelos previos en la literatura omiten. Esto permite mejorar la calidad de los turnos e itinerarios que pertenecen a la solución del VCSP.

Con el propósito de evaluar el desempeño de la formulación propuesta, se presentó un caso de estudio que resolvió, para cada día de la semana y línea de buses de la red, instancias reales de uno de los principales operadores de buses de Transantiago, Chile. En particular, se propusieron dos experimentos para evaluar el

trade-off entre la posibilidad de priorizar la minimización de turnos y la posibilidad de priorizar la minimización de itinerarios.

Respecto a la estructura de las soluciones, los resultados mostraron que cuando se prioriza la minimización de turnos es posible incurrir en un déficit de buses en algunas subredes. Por otra parte, cuando se prioriza la minimización de itinerarios hay un aumento significativo en los requerimientos de turnos de conducción. Estos resultados indican que debe existir una solución intermedia que entregue un equilibrio entre ambos objetivos. Esta solución intermedia dependerá principalmente de dos elementos de gestión empresarial: (i) Cobertura mínima de viajes que debe cubrir el operador para no recibir penalizaciones económicas del organismo regulador, y (ii) Disponibilidad diaria real de buses y conductores.

Respecto a los tiempos de solución en nuestro caso de estudio, los resultados mostraron que cuando se prioriza la minimización de itinerarios, los tiempos de solución son comparativamente mucho más altos en relación con los tiempos de solución obtenidos cuando se prioriza la minimización de turnos. Sin embargo, en términos absolutos, estos tiempos de solución son bajos, aun cuando el tamaño y la complejidad de las instancias probadas fueron grandes.

REFERENCIAS

- Bunte, S. & Kliwer, N. (2009). An Overview on Vehicle Scheduling Models. *Public Transport* 1, 299-317.
- Ceder A. (2002). Urban Transit Scheduling: Framework, Review and Examples. *Journal of Urban Planning and Development*, 128, 225-244.
- Freling, R.; Huisman, D. & Wagelmans, A. (2003). Models and algorithms for integration of vehicle and crew scheduling. *Journal of Scheduling*, 6(1), 63-85.
- Gaffi, A. & Nonato, M. (1999). An integrated approach to extra-urban crew and vehicle scheduling. In: Wilson, N. (Ed), *Computer-Aided Transit Scheduling. Lecture notes in economics and mathematical systems. Springer Verlag, Berlin, Germany*, 471,103-128.
- Groot, S. W. & Huisman, D. (2008). Vehicle and crew scheduling: solving large real-world instances with an integrated approach. *Lecture Notes in Economics and Mathematical Systems* 600, 43-56.
- Guihaire, V. & Hao, J.K. (2008). Transit network design and scheduling: a global review. *Transportation Research Part A: Policy and Practice*, 42(10), 1251-1273.
- Haase, K. & Friberg, C. (1999). An exact branch and cut algorithm for the vehicle and crew scheduling problem. In: Wilson, N. (Ed), *Computer-Aided Transit Scheduling. Lecture notes in economics and mathematical systems. Springer Verlag, Berlin, Germany*, 471, 63-80
- Haase, K.; Desaulniers, G. & Desrosiers, J. (2001). Simultaneous vehicle and crew scheduling in urban mass transit systems. *Transportation Science* 35(3), 286-303.
- Huisman, D.; Freling R., & Wagelmans A. P. (2005). Multiple-depot integrated vehicle and crew scheduling. *Transportation Science*, 39(4), 491-502.
- Kliwer, N.; Amberg, B. & Amberg, B. (2012). Multiple depot vehicle and crew scheduling with time windows for scheduled trips. *Public Transport*, 3, 213-244.
- Laurent, B. & Hao, J. K. (2008). Simultaneous vehicle and crew scheduling for extra urban transports. *Lecture Notes in Computer Science*, 5027, 466-475.
- Mesquita, M. & Paias, A. (2008). Set partitioning/covering-based approaches for the integrated vehicle and crew scheduling problem. *Computers & Operations Research*, 35, 1562-1575.
- Rodrigues, M. M.; Souza, C. C. & Moura, A. V. (2006). Vehicle and crew scheduling for urban bus lines. *European Journal of Operational Research*, 170(3), 844-862.