



UNIVERSIDAD DE BUENOS AIRES  
FACULTAD DE CIENCIAS EXACTAS Y NATURALES  
DEPARTAMENTO DE COMPUTACIÓN

# Modelos de programación lineal entera para la gestión de interrupciones ferroviarias en tiempo real con reprogramación de tripulaciones

Tesis presentada para optar al título de  
Licenciado en Ciencias de la Computación

Gastón Requeni

Director: Juan José Miranda Bront (UTDT / CONICET, [jmiranda@utdt.edu](mailto:jmiranda@utdt.edu))

Buenos Aires, Junio 2021

# Resumen

Grandes interrupciones ocurren a diario en las redes ferroviarias. A fin de dar una rápida respuesta a los usuarios, se dispone de un corto período de tiempo para reprogramar servicios y mantener la red en funcionamiento con la menor cantidad de cancelaciones y demoras posibles. Este problema se suele dividir en tres partes: reprogramación de cronograma, de material rodante y de tripulación. En la literatura se encuentran diversos modelos de programación lineal entera para resolver cada uno de los subproblemas y para resolver el problema completo se utiliza un algoritmo que resuelve cada parte de manera secuencial e iterativa hasta converger a una solución en común. El modelo de Veelenturf et al. [1] resuelve el problema de reprogramación de cronograma en una ventana de tiempo incorporando restricciones de material rodante, con el objetivo de mejorar la factibilidad del segundo subproblema y de esa manera mejorar la calidad de la solución final. En este trabajo se extiende dicho modelo para incorporar algunas restricciones de tripulación, con el objetivo de mejorar también la factibilidad en el tercer subproblema y por lo tanto mejorar aún más la calidad de la solución final. Se pone a prueba este modelo en instancias de la red ferroviaria del tren Mitre acotada al recorrido Retiro-Tigre y se compara con un modelo secuencial similar al utilizado en la literatura. Se observa una mejora considerable en la calidad de las soluciones obtenidas, especialmente en cuanto al uso de la tripulación.

**Palabras claves:** Gestión de interrupciones, Reprogramación de cronograma ferroviario, Reprogramación de tripulación, Programación Lineal Entera.

# Índice general

<b>1. Introducción</b>	5
<b>2. Reprogramación de cronograma ferroviario</b>	8
2.1. Descripción general del problema	8
2.1.1. Programación original y restricciones generales	8
2.1.2. Recursos necesarios para el funcionamiento del servicio	9
2.1.3. Bloqueo de vías	10
2.2. Red Evento-Actividad	11
2.2.1. Conjuntos de eventos	11
2.2.2. Conjuntos de actividades	11
2.3. Formulación PLE	13
2.4. Actividades de inventario	15
2.4.1. Vías de tramos	15
2.4.2. Vías de estación	16
2.4.3. Material Rodante	18
2.5. Bloqueo de una sección de vías	19
2.5.1. Trenes en el área del bloqueo	20
2.5.2. Eventos incluidos en el modelo	21
<b>3. Modelo integrado de reasignación de tripulación</b>	22
3.1. Problema de reasignación de tripulación	22
3.2. Modelo matemático	24
3.2.1. Tiempos de conexión	25
3.2.2. Grafo de alcance	25
3.2.3. Red de flujo de tripulación	28
3.3. Red de Evento-Actividad Extendida	30
3.4. Modelo PLE	31
3.4.1. Restricciones del recurso de tripulación	32
3.4.2. Función Objetivo	33
3.4.3. Horarios de almuerzo	36
3.5. Bloqueo de vías	39
3.6. Recorte temporal a la red de flujo de tripulación	41
3.6.1. Tareas previas al inicio del bloqueo	41
3.6.2. Tripulaciones que deben ser reposicionadas	44
3.6.3. Tareas posteriores al fin del bloqueo	45
3.6.4. Compleción de jornadas	47
3.7. Mejoras a la factibilidad	49
3.7.1. Reposicionamientos en taxi	49
3.7.2. Horas extra	50

---

3.7.3. Omitir almuerzo . . . . .	52
<b>4. Resultados preliminares . . . . .</b>	<b>53</b>
4.1. Caso de estudio: Línea Mitre . . . . .	53
4.1.1. Ampliación a la cantidad de tareas por servicio . . . . .	55
4.2. Parámetros . . . . .	56
4.3. Penalizaciones . . . . .	58
4.4. Implementaciones integrada y secuencial . . . . .	60
4.4.1. Parámetros de ejecución . . . . .	60
4.5. Métricas de calidad . . . . .	61
4.6. Análisis de resultados . . . . .	62
4.6.1. Análisis de factibilidad . . . . .	62
4.6.2. Análisis de calidad . . . . .	64
<b>5. Conclusiones y trabajo futuro . . . . .</b>	<b>73</b>
5.1. Conclusiones . . . . .	73
5.2. Trabajo futuro . . . . .	74
5.2.1. Experimentación . . . . .	74
5.2.2. Modelo matemático . . . . .	74
5.2.3. Algoritmo . . . . .	75

# 1. Introducción

La operación de redes ferroviarias para transporte de pasajeros afronta diariamente inconvenientes como el mal funcionamiento de la infraestructura, accidentes, avería de material rodante, paros sindicales, etc. Estos inconvenientes generan grandes interrupciones al normal funcionamiento de la red, reduciendo temporalmente la capacidad de la misma, provocando demoras y cancelaciones en los servicios. Esto naturalmente impacta en la calidad del servicio y provoca que los pasajeros lo utilicen menos frecuentemente, generando una mayor afluencia de tránsito en las calles, lo cual genera embotellamientos, contaminación sonora y ambiental. Los trenes son considerados en muchos países un medio de transporte ecológico y por lo tanto se desea promover su uso. Esto motiva la necesidad de minimizar el impacto que tienen las grandes interrupciones en la calidad del servicio.

La *gestión de grandes interrupciones* busca resolver el problema de reprogramar los horarios y rutas de los servicios, y reasignar todos los recursos necesarios para operarlos, durante una reducción en la capacidad de la red a causa de un bloqueo, minimizando el impacto en la calidad del servicio (principalmente las cancelaciones). Se dispone de un corto período de tiempo para encontrar una solución ya que el estado de la red cambia continuamente. Como se explica en Dollevoet et al. [3], debido a su complejidad y necesidad de resolverlo en tiempo real, se utiliza un **framework iterativo**, que lo parte en tres subproblemas que se resuelven de manera secuencial e iterativa (ver Figura 1.1): reprogramación de cronograma, reasignación de material rodante y reasignación de tripulación.

Como se indica en Veelenturf et al. [6], resolverlos de forma integrada usualmente no es una alternativa viable. En Potthoff [5] se menciona el trabajo de Walker [11] cómo la única aproximación conocida a resolverlos de manera integrada, pero únicamente considerando reprogramación del cronograma y de la tripulación, dejando fuera el material rodante. Se realizan pruebas sobre una red lineal pequeña de tan sólo 36 servicios de tren y dado que no se realizan recortes temporales o espaciales a la red, es decir se reprograma el día completo de toda la red, se menciona que esta idea probablemente no escale a redes de mayor tamaño y haya que reprogramar en una ventana de tiempo.

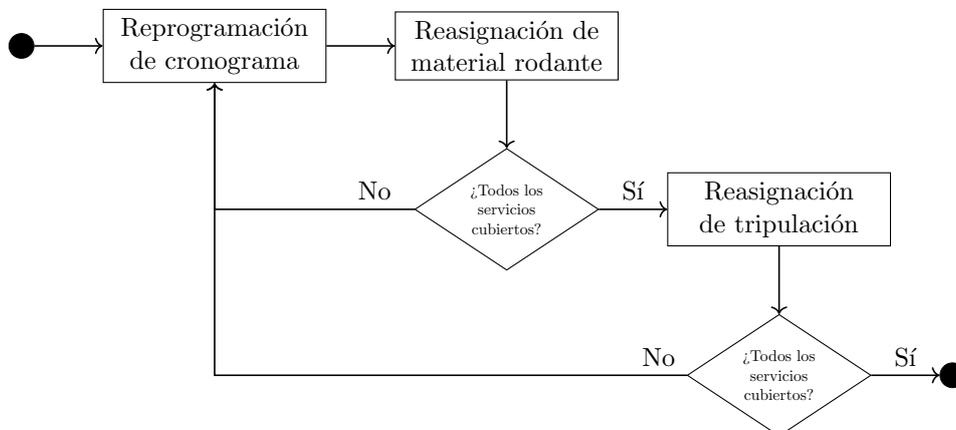


Fig. 1.1: Esquema del framework iterativo.

La primera fase del framework es la reprogramación de cronograma ferroviario, que consiste en determinar nuevos horarios de arribos y salidas de los trenes en cada una de las estaciones del recorrido, minimizando demoras y cancelaciones. Se divide

en dos niveles de abstracción: macroscópico y microscópico. El nivel macroscópico abstrae detalles físicos de la red y se concentra en el uso de recursos compartidos (como vías y estaciones), horarios y rutas, mientras que el nivel microscópico tiene en cuenta los detalles físicos de la red como pueden ser las intersecciones de vías, secciones de bloqueo, etc. Dado que el problema de gestión de grandes interrupciones es complejo, la reprogramación del cronograma se suele resolver a nivel macroscópico y luego se adapta la solución a la realidad (nivel microscópico) agregando pequeñas demoras que no afectan la asignación de recursos.

Las demoras y cancelaciones introducidas en la primera fase afectan a la disponibilidad y asignación original de material rodante, por lo tanto en la segunda fase se realiza una reasignación de material rodante para satisfacer los nuevos horarios. Si hay servicios que no se pueden cubrir, se vuelve a la primera fase con esa información y se genera un nuevo cronograma con la intención de que pueda ser cubierto por material rodante.

Una vez que todos los servicios pudieron ser cubiertos por material rodante, se pasa a la fase de reasignación de tripulación. Tanto las demoras y cancelaciones de la primera fase como las reasignaciones de material rodante de la segunda, afectan a la asignación original de las tripulaciones a los servicios. Por un lado, los cambios del cronograma y cancelaciones modifican la disponibilidad de tripulación para operar los servicios desde las estaciones designadas, y por otro lado la cantidad y tipos de vagones de un servicio impacta en la tripulación requerida. Nuevamente si no se encuentra una reasignación de tripulación que cubra todos los servicios, se vuelve a la primera fase.

Este proceso se repite hasta que los tres subproblemas encuentren una solución factible. Para reducir la cantidad de iteraciones y asegurar la convergencia, en Dollevoet et al. [3] se utilizan dos algoritmos para resolver la fase de reprogramación de cronograma:

- un algoritmo sofisticado para calcular la primer reprogramación de horarios, y
- un algoritmo goloso más simple para reaccionar ante infactibilidad del problema de material rodante o tripulación.

Para el algoritmo sofisticado se utilizó Veelenturf et al. [1], el cual utiliza una ventana de tiempo para reprogramar los servicios de tren, con la ventaja de que incluye algunas restricciones simplificadas de material rodante y de esta manera disminuye las cancelaciones adicionales por infactibilidad de la asignación de material rodante. Además disminuye la cantidad de iteraciones necesarias entre las dos primeras partes del framework.

Para el algoritmo goloso se utilizó uno bien simple: cancelar todos los servicios que no fueron cubiertos en la reasignación de material rodante o tripulación. Es decir que si alguna de esas fases no pudo cubrir un servicio, al volver a la primera fase dicho servicio se cancela.

Finalmente poniendo a prueba el framework, los resultados mostraron que la fase de reasignación de tripulación generaba muchas cancelaciones adicionales, dependiendo principalmente del horario en que se produce el bloqueo. Esto se debe a que cada tripulación tiene un horario y hay ciertos horarios de recambio de jornadas, en las que hay menos cantidad de tripulantes para asignar. En general dos iteraciones completas del framework bastaron para encontrar una solución factible en la mayoría de las instancias, debido al algoritmo goloso utilizado en la primera fase que cancela los servicios que no se pueden cubrir.

Como se menciona en Veelenturf et al. [6], cuando hay 1000 pasajeros esperando en una estación durante hora pico, siempre será preferible demorar el servicio antes que cancelarlo. Esto motiva buscar una mejora al framework para reducir los servicios que se cancelan por no poder cubrirse en la última fase. Como mencionamos

---

previamente, en Veelenturf et al. [1] se introdujeron algunas restricciones de material rodante en la primera fase para disminuir cancelaciones a causa de la fase de material rodante, con lo cual en este trabajo analizamos la posibilidad de incluir también algunas restricciones de tripulación en la primera fase, extendiendo el modelo de Veelenturf et al. [1], y de esta manera reducir cancelaciones causadas por la tercera fase.

El trabajo se divide de la siguiente manera. En el Capítulo 2 estudiamos el modelo de Veelenturf et al. [1] y presentamos el modelo PLE en el que se basa nuestro trabajo. Luego construimos un modelo para el problema de reprogramación de tripulación en el Capítulo 3 integrándolo al modelo del Capítulo 2. En el Capítulo 4 realizamos un estudio experimental a fin de estimar la dificultad computacional y evaluar la calidad de la solución del problema integrado. Finalmente en el Capítulo 5 realizamos algunas conclusiones y mencionamos trabajo a futuro.

## 2. Reprogramación de cronograma ferroviario

En el presente capítulo realizamos una introducción al problema de reprogramación del cronograma ferroviario y presentamos una formulación PLE para abordarlo. Debido a que este modelo es la base de nuestro trabajo, omitiremos algunos detalles de implementación y casos particulares haciendo hincapié en la parte central. Más detalles de la construcción del modelo pueden encontrarse en Veelenturf et al. [1] y Mosteiro [2].

### 2.1. Descripción general del problema

Nuestro problema surge en el contexto de un servicio de trenes y una red física asociada (compuesta por estaciones, vías, puntos de cruce, entre otros) en la cual, a partir de una contingencia, se produce un **bloqueo** de algún tramo de vías de la red. Dicho bloqueo vuelve infactible la **programación** original del servicio de trenes. El objetivo del problema es recalcular el cronograma en tiempo real, minimizando el impacto en la calidad del servicio para los pasajeros. Para cumplir con este objetivo, el modelo debe decidir qué trenes son **cancelados** (total y/o parcialmente) y cuáles son **retrasados**, conformando un nuevo programa que cumpla con las restricciones operativas y de seguridad, que será utilizado hasta reestablecer el servicio normal.

Para lograr esto, debe tenerse en cuenta toda la información disponible, tanto del programa original como del estado actual de la red sobre la que operan los trenes y el bloqueo de vías. A lo largo del trabajo, asumimos que este es el caso así como también suponemos conocer una estimación de la duración total del bloqueo.

#### 2.1.1. Programación original y restricciones generales

Una red ferroviaria consiste de **estaciones** conectadas por **tramos de vías**. Cada estación y cada tramo pueden tener una o más vías. Algunas estaciones tienen un patio de maniobras, donde se almacenan composiciones de **material rodante**, que pueden ser utilizados para operar los trenes. En la Figura 2.1 presentamos un ejemplo ilustrativo de una red ferroviaria.

Un cronograma contiene la información del recorrido de cada uno de los trenes. Llamamos  $T$  al conjunto de todos los trenes. Para cada tren  $t \in T$ , el cronograma indica el **horario de salida** y el **horario de llegada** en cada una de las estaciones de su recorrido. Llamamos  $S$  al conjunto de todas las estaciones de la red. En la Tabla 2.1 observamos un posible cronograma para la red de la Figura 2.1.

El programa original es el cronograma planificado al inicio del día, antes de que ocurra el bloqueo. Es importante mencionar que los horarios originales deben ser considerados en el nuevo cronograma, dado que un tren no puede ser reprogramado en un tiempo anterior a su horario previo.

En cuanto a los **tiempos mínimos** de circulación entre dos estaciones y de espera dentro de una estación, los mismos podrían variar respecto del programa original. Por ejemplo, en caso de una sección afectada por el bloqueo, los tiempos de viaje en el nuevo cronograma pueden ser mayores a los tiempos de viaje originales en dicha sección. Por otro lado, en otras secciones se podrían permitir tiempos de viaje

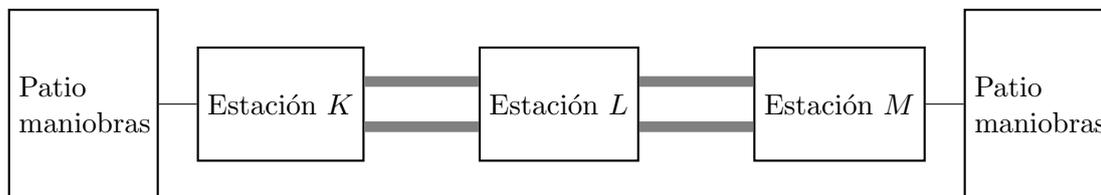


Fig. 2.1: Ejemplo de red ferroviaria

Servicio	Salida $K$	Arribo $L$	Salida $L$	Arribo $M$
Tren 1	7:00	7:20	7:23	7:30
Tren 2	7:55	8:15	8:18	8:25

Servicio	Salida $M$	Arribo $L$	Salida $L$	Arribo $K$
Tren 3	7:16	7:23	7:26	7:46
Tren 4	7:45	7:52	7:55	8:19

Tab. 2.1: Ejemplo de cronograma

o espera menores a los originales, siempre y cuando no se viole ninguna restricción de seguridad.

Otro dato a tener en cuenta es el **tiempo máximo de retraso** que un tren puede tener en cada salida o llegada de una estación determinada. Este parámetro define cuánto estamos dispuestos a deteriorar el servicio demorando trenes, es decir cuál es la demora máxima permitida en el horario de cada evento. Es de suma importancia ya que en caso de permitir un retraso máximo bajo, se limitan las opciones a considerar, forzando a que se produzcan muchas cancelaciones. Por otro lado, si los horarios originales pueden ser retrasados por mucho tiempo, esto implicará que existan muchas opciones a considerar, lo cual podría tener un impacto muy grande en los tiempos de cómputo.

A la hora de recalculer el programa original, debe establecerse una relación entre los retrasos y las cancelaciones. Un tren cancelado debería tener un impacto negativo mucho mayor que una llegada o salida retrasada, sin embargo, podría ser preferible tener unas pocas cancelaciones que eviten una gran cantidad de retrasos. Por dicho motivo, se incluye en el modelo una **penalización por tren cancelado** y una **penalización por unidad de retraso** que modelan esta relación.

### 2.1.2. Recursos necesarios para el funcionamiento del servicio

Para que los servicios puedan ser operados, debe garantizarse la disponibilidad de ciertos recursos. Como mencionamos previamente, los recursos presentes en una red ferroviaria son las estaciones, los tramos de vías y el material rodante. Un servicio de tren deberá utilizar estos recursos para operar, con lo cual limitan las posibles reprogramaciones de horarios. Cada servicio usa a lo sumo un recurso de cada tipo en cualquier momento dado, es decir ocupa una estación o un tramo de vías, y utiliza una composición de material rodante (que forma físicamente el tren). La cantidad disponible de cada recurso se llama **capacidad**.

El recurso de **vías de tramo** tiene como capacidad la cantidad de vías disponibles. Cada vía puede ser usada en ambas direcciones, pero no al mismo tiempo si los trenes van en sentido contrario. Sin embargo puede haber varios servicios utilizando la misma vía si van en la misma dirección. Asumimos que un servicio no puede cambiar de vía en un tramo una vez que comienza a transitarlo, pero por el contrario tiene total libertad para cambiar de vías en la unión con las vías de estación (en

los extremos). Se deben respetar **tiempos de seguridad** entre los servicios, tanto entre dos servicios que van en la misma vía en la misma dirección, como entre dos servicios que usan la vía en sentidos opuestos. Estos tiempos son una abstracción de la separación requerida entre los distintos trenes para evitar colisiones.

El recurso de **vías de estación** también tiene como capacidad la cantidad de vías disponibles y las vías pueden ser usadas en ambas direcciones. Pero a diferencia de las vías de tramo, sólo un servicio puede estar ocupando una vía a la vez. Se deben respetar tiempos de seguridad entre que un tren sale de la vía y otro ingresa. Como mencionamos previamente, algunas estaciones tienen patio de maniobras, los cuales asumimos que tienen capacidad infinita.

El recurso de **material rodante** se encuentra distribuido en los **patios de maniobras** al comienzo del día. Entonces en cada estación que tiene patio de maniobras, la capacidad de este recurso es la cantidad de composiciones de material rodante que se encuentran allí (trenes completos, no vagones aislados). Asumimos que estas composiciones no se pueden partir. Un servicio de tren utiliza el mismo material rodante de principio a fin, es decir que no permitiremos intercambios durante el recorrido. Cuando un servicio finaliza y libera el material rodante, este puede traspasarse a otro servicio o bien almacenarse en el patio de maniobras de la estación (si tiene). Al final del día, cada composición debe terminar en un patio de maniobras. Debe respetarse un tiempo mínimo de maniobra para traspasar material rodante de un servicio a otro. Por último, no cualquier material rodante puede ser utilizado por cualquier servicio, sino que puede haber tipos de material rodante y cada servicio requiere cierto tipo.

### 2.1.3. Bloqueo de vías

El bloqueo de una sección de vías vuelve infactible la programación original, siendo este el motivo que da origen a nuestro problema. El bloqueo ocurre en un **tramo** determinado, bloqueando una **cantidad** de vías durante un período de cierta **duración**.

El tramo de vías afectado queda determinado por el par de estaciones que conecta. En nuestro trabajo asumiremos que los bloqueos afectan una única sección para facilitar la experimentación. Sin embargo, el modelo es fácilmente adaptable para el caso de múltiples bloqueos en distintos tramos. En cuanto a los bloqueos de vías de estaciones, si bien no son considerados, pueden ser tratados de una manera similar a los bloqueos de vías de tramos.

En cuanto a la cantidad de vías bloqueadas, el bloqueo puede afectar solo algunas vías (bloqueo parcial) o el total de la capacidad del tramo (bloqueo total). Es importante notar que en caso de un bloqueo total, durante el transcurso del mismo ningún tren podrá circular por la sección afectada.

El último dato clave es la duración del bloqueo. Para esto, el modelo asume que no solo el **tiempo de inicio del bloqueo** es conocido sino también el **tiempo de fin del bloqueo**. Vale mencionar que en la práctica muchas veces se desconoce la duración del bloqueo, en estos casos, se trabaja preventivamente con una estimación del tiempo de fin que se puede ir actualizando a medida que se obtengan nuevos datos. Adicionalmente, se debe determinar un instante de tiempo posterior al bloqueo en el cual los trenes vuelven a funcionar de acuerdo a la programación original. Lo llamamos **tiempo límite de transición** dado que modela el instante en que los trenes dejan de seguir el nuevo cronograma y vuelven a estar programados bajo el cronograma original. Luego, el modelo debe asegurar que a partir de ese momento todos los trenes puedan circular sin ser retrasados ni cancelados. De esta manera, la elección de dicho instante de tiempo no es menor, dado que afectará las posibilidades a considerar para el armado de la nueva programación. La Figura 2.2 muestra los instantes mencionados en una línea de tiempo.

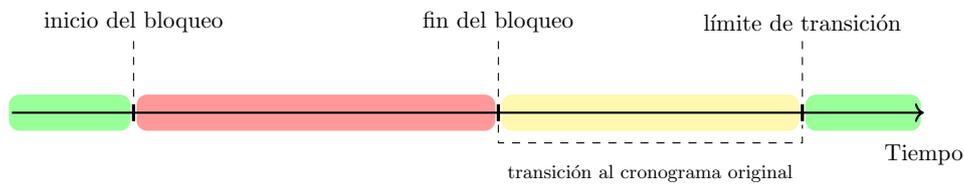


Fig. 2.2: Bloqueo y sus instantes de tiempo

Los trenes que estén programados para circular por la sección del bloqueo durante el transcurso del mismo deben tratarse con especial atención. Tanto el manejo de estos casos como el de varios otros detalles que deben tomarse para contemplar correctamente el bloqueo pueden verse en la Sección 2.5.1.

## 2.2. Red Evento-Actividad

Como mencionamos previamente, nuestro objetivo es encontrar un cronograma factible para la red con capacidad reducida. Esto implica determinar qué trenes cancelar, y los nuevos horarios de partida y arribo para los trenes restantes. Para esto vamos a representar a cada servicio como una secuencia de **eventos** de salida y arribo en las distintas estaciones del recorrido. El objetivo de la reprogramación se traduce en determinar el horario en el cual deberán ocurrir dichos eventos. Por otro lado, para modelar el traspaso de un recurso de un evento a otro utilizaremos **actividades**.

Una **red evento-actividad**  $N = (E, A)$  es un grafo dirigido, siendo  $E$  el conjunto de nodos (llamados eventos) y  $A$  el conjunto de ejes (llamados actividades). Es una red espacio tiempo ya que los eventos identifican una estación y un cierto horario, y además los ejes representan el traspaso de los recursos utilizados en dichos eventos.

### 2.2.1. Conjuntos de eventos

El conjunto de eventos  $E = E_{train} \cup E_{inv}$  consiste de un conjunto de **eventos de tren**  $E_{train}$  y un conjunto de **eventos de inventario**  $E_{inv}$ .

Cada evento de tren  $e \in E_{train}$  representa un arribo o una salida de un servicio de tren  $t_e$  en una estación específica. El tiempo en el que este evento estaba programado en el cronograma original es  $q_e$ . Además cada evento admite una demora máxima que llamaremos  $d_e$ . Los eventos de tren están asociados con un conjunto de recursos que necesitan para ocurrir.

Los eventos  $e \in E_{inv}$  representan el inventario de un recurso determinado al comienzo del día, siendo  $i_e$  la capacidad. Cada tramo de vías, cada estación y cada patio de maniobras tendrá un evento de inventario asociado a la cantidad de vías disponibles en los dos primeros casos y la cantidad de composiciones de material rodante al inicio del día en el último caso.

### 2.2.2. Conjuntos de actividades

Las actividades de la red representan el uso secuencial de cada recurso. Concretamente, el uso de una actividad  $a = (e, f) \in A$  dirigida del evento  $e \in E$  hacia el evento  $f \in E$  indica que  $f$  utiliza uno de los recursos que utilizó  $e$ , después de que  $e$  haya ocurrido. Entre cada par de eventos puede haber más de una actividad, pero no más de una por recurso. Cada actividad tiene asociada una mínima duración  $L_a$

que representa el tiempo mínimo desde la ocurrencia de  $e$  hasta que el recurso se encuentra disponible para  $f$ .

Para cada evento  $e \in E$ , dividimos en dos grupos las actividades asociadas: por un lado las actividades de entrada  $A^-(e) = \{a = (f, e) \in A | f \in E\}$  y por otro las actividades de salida  $A^+(e) = \{a = (e, f) \in A | f \in E\}$ .

Cada actividad  $(f, e)$  en  $A^-(e)$  se refiere a que  $e$  puede utilizar un recurso  $L_a$  minutos después de que ocurre  $f$ . Según el tipo de recurso, distinguiremos tres subconjuntos: (i)  $A_{track}^-(e) \subset A^-(e)$  para las vías de tramo, (ii)  $A_{stat}^-(e) \subset A^-(e)$  para las vías de estación, y (iii)  $A_{rol}^-(e) \subset A^-(e)$  para el material rodante.

Por otro lado, cada actividad  $(e, f)$  en  $A^+(e)$  se refiere a que  $f$  puede utilizar un recurso  $L_a$  minutos después de que ocurre  $e$ . Análogamente distinguiremos los tres subconjuntos según el tipo de recurso:  $A_{track}^+(e)$ ,  $A_{stat}^+(e)$  y  $A_{rol}^+(e)$ .

Para referirnos de manera genérica a los subconjuntos de actividades de cada tipo de recurso utilizaremos los conjuntos de conjuntos  $\mathcal{A}^-(e) = \{A_{track}^-(e), A_{stat}^-(e), A_{rol}^-(e)\}$  y  $\mathcal{A}^+(e) = \{A_{track}^+(e), A_{stat}^+(e), A_{rol}^+(e)\}$ . Cabe aclarar que algunos eventos podrían no necesitar alguno de los recursos para ocurrir, y en tal caso el subconjunto correspondiente será vacío.

Como mencionamos previamente, cada servicio de tren debe permanecer un tiempo mínimo en cada estación y necesita un tiempo mínimo para recorrer cada tramo de vías. Para modelar estos tiempos mínimos también utilizaremos actividades, que llamaremos **actividades de tren**, representadas con el conjunto  $A_{train}$ . La principal diferencia con las demás actividades (a las cuales nos referiremos convenientemente como **actividades de inventario**) es que unen a los eventos de un mismo servicio de tren en el orden que deben ocurrir.

Para mayor claridad, en la Figura 2.3 se pueden observar los eventos y actividades de tren asociados a la red de la Figura 2.1 y el cronograma de la Tabla 2.1. En secciones posteriores usaremos este mismo ejemplo para explicar la construcción de los conjuntos de actividades de inventario.

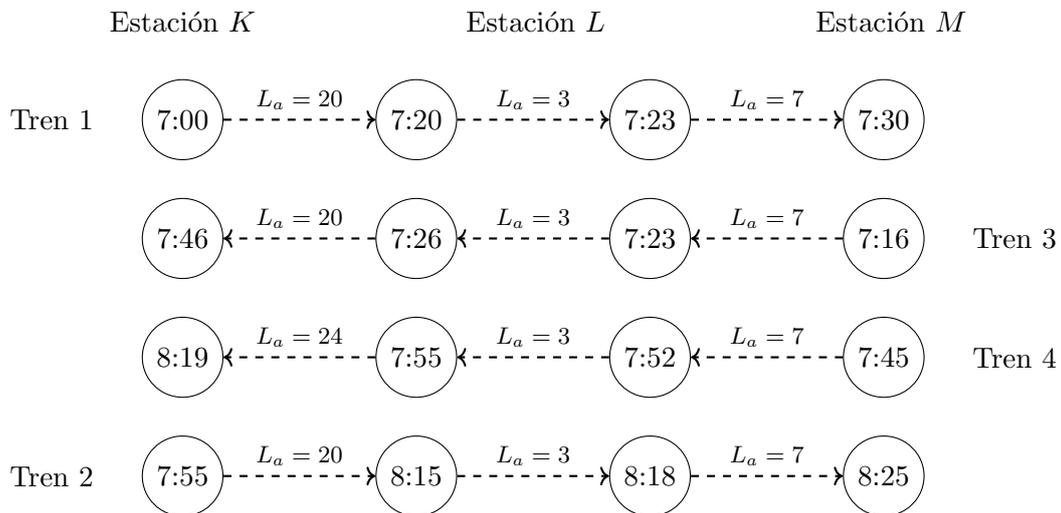


Fig. 2.3: Eventos y actividades de tren para el cronograma 2.1. Los eventos se organizan de manera horizontal según la estación y de manera vertical según el comienzo de cada servicio de tren. Los valores en cada evento son sus horarios originales (i.e.  $q_e$ ).

### 2.3. Formulación PLE

A continuación se presenta el modelo de PLE, el cual describe la función objetivo a minimizar y las restricciones que deben cumplir los nuevos tiempos reprogramados para los eventos de cada tren. Para modelar la posible cancelación de cada tren  $t \in T$ , definimos las variables binarias  $y_t$  que toman valor 1 si y sólo si el tren  $t \in T$  se cancela. Llamamos  $\lambda_t$  a la penalización por cancelar el tren. Por otro lado, para representar el horario de un evento de tren  $e \in E_{train}$  en el nuevo cronograma, definimos la variable entera  $x_e$ , y llamamos  $\mu_e$  a la penalización por demorar dicho evento. Por último, para determinar la asignación de recursos a cada evento debemos elegir un subconjunto de actividades de la red que modele el traspaso secuencial de cada uno de ellos. Para esto utilizaremos las variables binarias  $z_a$  para cada actividad de inventario  $a \in A_{inv}$ , que indican si dicha actividad será utilizada para traspasar un recurso en la reprogramación.

Junto con el modelo, se agrega la Tabla 2.2 que resume la notación utilizada. A pesar de que aún no definimos las actividades de cada recurso, introduciremos las restricciones que estas actividades deberán cumplir y luego veremos en detalle cómo se interpretan según cada recurso.

$$\text{Min} \quad \sum_{t \in T} \lambda_t y_t + \sum_{e \in E_{train}} \mu_e (x_e - q_e) \quad (2.1)$$

$$\text{s.a.} \quad x_e - q_e \geq 0 \quad \forall e \in E_{train} \quad (2.2)$$

$$x_e - q_e \leq (1 - y_{t_e}) d_e \quad \forall e \in E_{train} \quad (2.3)$$

$$x_f - x_e \geq L_a \quad \forall a = (e, f) \in A_{train} \quad (2.4)$$

$$\sum_{a \in C} z_a + y_{t_e} = 1 \quad \forall e \in E_{train}, C \in \mathcal{A}^-(e), C \neq \emptyset \quad (2.5)$$

$$\sum_{a \in C} z_a + y_{t_e} \leq 1 \quad \forall e \in E_{train}, C \in \mathcal{A}^+(e) \quad (2.6)$$

$$\sum_{a \in C} z_a \leq i_e \quad \forall e \in E_{inv}, C \in \mathcal{A}^+(e) \quad (2.7)$$

$$x_f - x_e + M_a(1 - z_a) \geq L_a \quad \forall a = (e, f) \in A_{inv}, e, f \in E_{train} \quad (2.8)$$

$$y_t \in \{0, 1\}, x_e \in \mathbb{N}, z_a \in \{0, 1\} \quad \forall t \in T, \forall e \in E_{train}, \forall a \in A \quad (2.9)$$

La función (2.1) modela nuestro objetivo de encontrar una reprogramación buscando minimizar el impacto en la calidad de servicio. Para esto, por un lado se suma una penalización por cada tren cancelado y por otro lado se suma una penalización por cada unidad de retraso de un evento en la nueva programación. Luego, si un tren es reprogramado en el mismo horario que lo estaba originalmente, su impacto será nulo. El modelo debe buscar la mínima cantidad de cancelaciones y retrasos dentro de las posibilidades que brinda la red al estar limitada por el bloqueo.

Las restricciones (2.2) aseguran que los eventos no puedan ser reprogramados antes de su horario original. Asimismo, las restricciones (2.3) aseguran que los eventos respeten el límite máximo de retraso en caso de que sus respectivos trenes no sean cancelados. Vale notar que en caso de que un tren sea cancelado, el modelo fuerza a que el nuevo horario de sus eventos sea igual al horario original. Esto último es importante para que no se pese la penalización por retraso de sus eventos sino solamente por la penalización por la cancelación del tren. Si bien una solución óptima

$T$		conjunto de trenes
$y_t$	variable de decisión de cancelación del tren $t \in T$ (1 = cancelado)	
$\lambda_t$		penalización por cancelación del tren $t \in T$
$E$		conjunto de eventos
$E_{train}$		eventos de tren
$E_{inv}$		eventos de inventario
$t_e$		tren correspondiente al evento $e \in E_{train}$
$q_e$		horario original del evento $e \in E_{train}$
$x_e$	variable que determina el nuevo horario del evento $e \in E_{train}$	
$d_e$		retraso máximo permitido para el evento $e \in E_{train}$
$\mu_e$		penalización por unidad de retraso del evento $e \in E_{train}$
$i_e$	cantidad de recursos disponibles por el evento de inventario $e \in E_{inv}$	
$A$		conjunto de todas las actividades
$z_a$	variable de decisión de selección de la actividad $a \in A_{inv}$ (1 = seleccionada)	
$L_a$		duración mínima de la actividad $a \in A$
$A_{train}$		actividades de tren
$A_{inv}$		conjunto de actividades de inventario
$\mathcal{A}^-(e)$		el conjunto $\{A_{track}^-(e), A_{stat}^-(e), A_{rol}^-(e)\}$
$\mathcal{A}^+(e)$		el conjunto $\{A_{track}^+(e), A_{stat}^+(e), A_{rol}^+(e)\}$

Tab. 2.2: Notación formulación PLE base

forzaría esto mismo sin necesidad de incluir la restricción explícitamente, al hacerlo estamos limitando el espacio de posibles soluciones.

Las desigualdades (2.4) fuerzan a que se respeten los tiempos mínimos de viaje entre estaciones y de espera en una estación. En caso de que se utilicen como dichos valores las duraciones del cronograma original, si una actividad  $a = (e, f)$  conecta los eventos  $e$  y  $f$  y el evento  $e$  es retrasado en el nuevo programa (i.e.  $x_e > q_e$ ), el evento  $f$  también deberá retrasarse para que el tren pueda cumplir con la nueva programación.

Las igualdades (2.5) modelan cómo los eventos toman sus recursos necesarios para circular. Un evento debe tomar una unidad de cada recurso a través de una actividad o, en caso de no poder hacerlo, su tren debe ser cancelado. Es importante notar que el conjunto de igualdades fuerza que los eventos tomen un único recurso de cada tipo y no más. Otro detalle a tener en cuenta es que, en caso de que un evento no tenga actividades de entrada para un tipo de recurso (porque el evento no necesita tomar recursos de ese tipo), la igualdad no se define, dado que forzaría a cancelar el tren asociado al evento. Esto último queda más claro al estudiar cómo se definen los conjuntos de entrada para cada recurso, como veremos en las siguientes secciones.

Las desigualdades (2.6) modelan cómo los eventos traspasan recursos. Para cada tipo de recurso, un evento puede traspasar el recurso utilizado a lo sumo a otro evento. Al tratarse de una desigualdad en este caso, podría pasar que un evento no traspase sus recursos utilizados, modelando así el caso del último evento que utiliza un recurso determinado. También vale notar como en caso de que el tren del evento sea cancelado, no puede traspasar ningún recurso, esto tiene sentido dado que al ser cancelado, el evento tampoco pudo haber tomado ningún recurso por las igualdades (2.5).

Las desigualdades (2.7) modelan el primer uso de cada unidad de recurso a través de los eventos de inventario. Como se explicó en secciones anteriores, los eventos de inventario modelan la capacidad de cada uno de los recursos disponibles en la red. Las actividades de salida desde estos eventos determinan cuales son los posibles eventos de tren que pueden tomar estos recursos por primera vez para luego ir traspasándolos (como marcan las restricciones (2.5) y (2.6)) hacia otros eventos.

Un evento de inventario  $e \in E_{inv}$  posee una capacidad  $i_e$  de recursos disponibles, las desigualdades (2.7) fuerzan a que los eventos de inventario no den más recursos de los que tienen disponibles. Es importante notar que los eventos de inventario no tienen actividades de entrada (i.e.  $A^-(e) = \emptyset, \forall e \in E_{inv}$ ), su función es la de brindar los recursos disponibles de la red a los primeros eventos que los necesiten. Más aún, como los eventos de inventario corresponden a un tipo de recurso determinado, solo poseen actividades de salida para ese tipo de recurso, para los otros tipos de recurso restantes sus conjuntos de salida son vacíos.

Las desigualdades (2.8) aseguran que se cumplan los tiempos de seguridad de los trasposos de recursos. Notar también que se está forzando un orden temporal entre los eventos, es decir, si  $e$  le trasposa el recurso a  $f$ , entonces  $e$  es programado antes que  $f$  en el nuevo cronograma. Las desigualdades son similares a las (2.4) que modelan los tiempos mínimos de viaje y de espera, la diferencia es que deben tener en cuenta el hecho de que la actividad  $a = (e, f)$  puede no ser utilizada (el evento  $f$  puede tomar el recurso de otro evento distinto de  $e$ ). Para estos casos, al tener  $z_a = 0$ , un valor de  $M$  suficientemente grande hace que la desigualdad no imponga condiciones. Dicho valor depende de la actividad  $a$ , y por ese motivo notamos a la cota  $M_a$ . Se puede ver fácilmente que tomando  $M_a = L_a + (q_e + d_e) - q_f$  es suficiente para que la desigualdad sea válida.

Finalmente, las desigualdades (2.9) modelan las variables utilizadas como binarias y enteras según corresponda.

## 2.4. Actividades de inventario

A continuación definimos los conjuntos de actividades de cada uno de los distintos recursos.

### 2.4.1. Vías de tramos

Cada actividad de este recurso representa el primer uso o el uso secuencial de una vía de tramo. El tramo de vías entre las estaciones  $k$  y  $l$  se representa con un par  $(k, l)$ .

Sea  $E_{dep}^k \subseteq E_{train}$  el conjunto de eventos de salidas de trenes de la estación  $k$ , y sea  $E_{dep}^{kl} \subseteq E_{dep}^k$  el conjunto de eventos de salida de la estación  $k$  de servicios cuya próxima estación es  $l$  (es decir que recorren el tramo  $(k, l)$  luego del evento). Similarmente definimos  $E_{arr}^l \subseteq E_{train}$  como los arribos a la estación  $l$  y  $E_{arr}^{kl} \subseteq E_{arr}^l$  como los arribos a la estación  $l$  de servicios cuya estación anterior fue  $k$ . Por último, definimos  $e_{track}^{kl} \in E_{inv}$  como el evento correspondiente al inventario inicial de vías en el tramo  $(k, l)$ , cuya capacidad  $i_{e_{track}^{kl}}$  equivale a la cantidad de vías en dicho tramo.

Para los eventos de cada estación  $k$  y  $l$  definimos los conjuntos  $A_{track}^-$  y  $A_{track}^+$ .

- Para una salida  $e \in E_{dep}^{kl}$  de la estación  $k$  hacia el tramo  $(k, l)$ :
  - El conjunto  $A_{track}^-(e)$  está formado por actividades  $a = (f, e)$  desde:
    1. Otras salidas en la misma dirección:  $f \in E_{dep}^{kl} \setminus \{e\}$ .
    2. Arribos en la dirección contraria:  $f \in E_{arr}^{lk}$ .
    3. El evento de inventario  $e_{track}^{kl}$ .
  - El conjunto  $A_{track}^+(e)$  está formado por actividades  $a = (e, f)$  hacia:
    1. Otras salidas en la misma dirección:  $f \in E_{dep}^{kl} \setminus \{e\}$ .
- Para un arribo  $e \in E_{arr}^{kl}$  a la estación  $l$  desde el tramo  $(k, l)$ :

- El conjunto  $A_{track}^-(e)$  es vacío, dado que un arribo a una estación no necesita tomar un recurso de vías de un tramo para ocurrir (el tren ya está en la vía).
- El conjunto  $A_{track}^+(e)$  está formado por actividades  $a = (e, f)$  hacia:
  1. Otros arribos en la misma dirección:  $f \in E_{arr}^{kl} \setminus \{e\}$ .
  2. Salidas en la dirección contraria:  $f \in E_{dep}^{lk}$ .
- Para el evento de inventario del tramo  $(k, l)$ , el conjunto  $A_{track}^+(e_{track}^{kl})$  está formado por actividades  $(e_{track}^{kl}, e)$  hacia salidas de ese tramo en ambas direcciones, es decir  $e \in E_{dep}^{kl} \cup E_{dep}^{lk}$ .

El recurso de vías de tramo es un requisito de los eventos de salida de una estación (i.e. ingreso al tramo). Este recurso lo podrán tomar del evento de inventario (primer uso del tramo), de una salida anterior en la misma dirección (un tren detrás del otro) o bien un arribo en la dirección contraria que libera completamente una vía del tramo.

Por otro lado, los eventos de arribo a una estación *implícitamente* poseen el recurso (por eso no necesitan tomarlo) y deben liberarlo al ingresar a la vía de estación. Pueden traspasar el recurso de la vía a una salida en la dirección contraria o bien a un arribo en la misma dirección (el tren que viene detrás).

Observar que en el caso de que un arribo traspase el recurso de la vía del tramo al arribo del tren que viene detrás genera una situación que viola la definición de grafo: la actividad  $(e, f)$  se encuentra entre las actividades que salen de  $e$  pero no entre las que ingresan a  $f$ . Esto sucede porque los arribos implícitamente poseen el recurso de la vía del tramo que acaban de recorrer. Supongamos que  $f$  tuviera la actividad de entrada. En caso de usarla, sucedería que  $f$  tendría dos vías de tramo para traspasar, la que recibió de  $e$  y la que implícitamente tenía por ser un arribo. Esta situación ocurre por el hecho de que podemos utilizar la vía por más de un tren a la vez en la misma dirección (más de una entidad utilizando una unidad del mismo recurso).

La Figura 2.4 muestra el grafo de eventos y actividades del recurso de vías de tramos para el tramo  $(K, L)$  de la red de ejemplo 2.1 y el cronograma asociado 2.1. Para facilitar el entendimiento, se incluyen también las actividades de tren en líneas punteadas. Además, si bien por la definición de los conjuntos el modelo permite que un evento traspase su recurso a un evento programado en un tiempo anterior, dichas actividades no son incluidas en la figura. Esto es así dado que estas actividades no podrían ser seleccionadas debido a las desigualdades (2.8), como mencionamos anteriormente. Para el evento de inventario, se incluye entre paréntesis su respectiva capacidad, en este caso, al tratarse del tramo  $(K, L)$  de la red ejemplo, la capacidad es de dos vías.

### 2.4.2. Vías de estación

Cada actividad de este recurso representa el primer uso o el uso secuencial de una vía de estación. Para definir las actividades nos basaremos en los conjuntos  $E_{dep}^k$  y  $E_{arr}^k$  definidos en la sección previa. Además definimos  $e_{stat}^k \in E_{inv}$  como el evento de inventario de vías de la estación  $k$ , siendo  $i_{e_{stat}^k}$  la cantidad de vías disponibles.

Para cada estación  $k$ :

- Para un arribo  $e \in E_{arr}^k$  en la estación  $k$ :
  - El conjunto  $A_{stat}^-(e)$  está formado por actividades  $a = (f, e)$  desde:
    1. Salidas desde la misma estación:  $f \in E_{dep}^k$ .
    2. El evento de inventario  $e_{stat}^k$ .

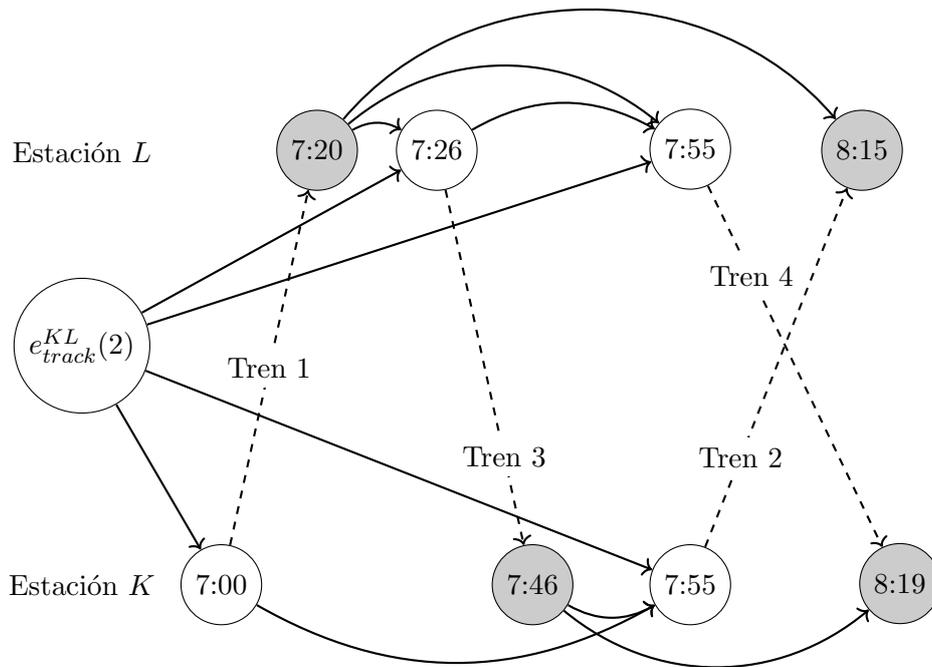


Fig. 2.4: Eventos y actividades relacionadas con el recurso de vías de tramo para el tramo  $(k, l)$  del cronograma ejemplo 2.1. Los eventos se organizan horizontalmente según su horario programado y verticalmente según su estación correspondiente. Los eventos correspondientes a salidas son blancos y los arribos grises. Los horarios están fijos para simplificar la ilustración, si tuviéramos en cuenta las demoras, nuevos traspasos del recurso serían posibles. Las actividades del recurso de vías de tramo se dibujan con un eje normal mientras que las actividades de tren se dibujan con una línea punteada. El evento de inventario del tramo de vías se incluye a la izquierda y entre paréntesis figura su capacidad.

- El conjunto  $A_{stat}^+(e)$  es vacío, dado que es claro que el próximo evento en utilizar la vía de la estación tomada por el arribo es su posterior salida. El traspaso se modela de manera implícita.
- Para una salida  $e \in E_{dep}^k$  desde la estación  $k$ :
  - El conjunto  $A_{stat}^-(e)$  es vacío por el mismo motivo mencionado arriba.
  - El conjunto  $A_{stat}^+(e)$  está formado por actividades  $a = (e, f)$  hacia:
    1. Otros arribos en la misma estación:  $f \in E_{arr}^k$ .
- Para el evento de inventario de la estación  $k$ , el conjunto  $A_{stat}^+(e_{stat}^k)$  está formado por actividades  $a = (e_{stat}^k, e)$  hacia arribos en la estación:  $e \in E_{arr}^k$ .

Intuitivamente los eventos de arribo a una estación requieren una vía de estación entonces deberán tomar el recurso del evento de inventario (en el primer uso) o bien lo tomarán de una salida de otro servicio que usó la vía previamente (uso secuencial). En este último caso  $L_a$  de la actividad asociada garantizará el tiempo de seguridad necesario entre los eventos de salida y arribo.

Por otro lado, los eventos de salida de la estación siempre cederán el recurso a un arribo posterior, a menos que no haya más arribos y en tal caso no es necesario cederlo (la restricción 2.6 no fuerza el traspaso, justamente por si no hay otro evento que necesite el recurso).

Para aquellas salidas  $e \in E_{dep}^k$  desde la estación  $k$  que correspondan al comienzo de un servicio de tren (el primer evento correspondiente al servicio), con las definiciones anteriores estaríamos asumiendo que el tren ya se encuentra en la vía de

estación, pero eso no es necesariamente así ya que podría encontrarse en el patio de maniobras antes de su primer evento. Entonces en estos casos especiales se toma  $A_{stat}^-(e)$  como si  $e$  fuera un arribo (es decir que requiere tomar el recurso del inventario o de una salida previa que libera la vía).

La Figura 2.5 muestra el grafo de eventos y actividades del recurso de vías de estación para la estación  $L$  de la red de ejemplo 2.1 y el cronograma asociado 2.1. Para facilitar el entendimiento, se incluyen también las actividades de tren en líneas punteadas. Al igual que en la Figura 2.4, si bien por la definición de los conjuntos el modelo permite que un evento traspase el recurso a un evento programado en un tiempo anterior, dichas actividades no son incluidas dado que no podrían ser seleccionadas por las desigualdades (2.8). Para el evento de inventario, se incluye entre paréntesis su respectiva capacidad, en este caso, al tratarse de la estación  $L$  de la red ejemplo, la capacidad es de dos vías. Por simplicidad, para el ejemplo se asume que los tiempos son fijos (no se permiten demoras), pero cabe aclarar que en la práctica los tiempos son variables y eso influye en el armado del grafo.

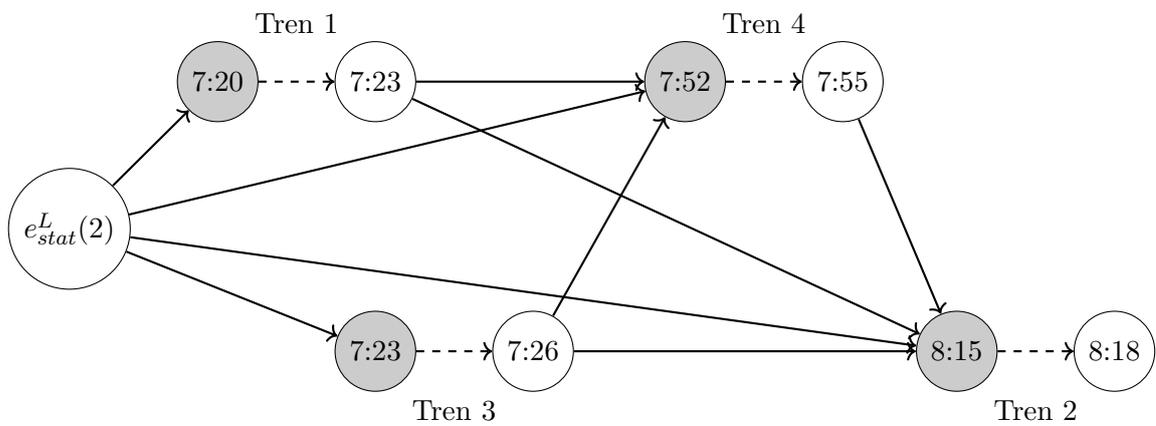


Fig. 2.5: Eventos y actividades relacionadas con el recurso de vías de una estación del cronograma ejemplo 2.1. Los eventos y actividades se organizan siguiendo la misma idea que la figura 2.4.

### 2.4.3. Material Rodante

Cada actividad de este recurso representa el traspaso de una composición completa de material rodante de un servicio de tren que finaliza a uno que comienza, o bien el primer uso del mismo. Como mencionamos en secciones anteriores, las composiciones no pueden partirse ni cambiar durante el trayecto de un tren.

Sea  $E_{start}^k \subset E_{train}$  el conjunto de los primeros eventos de salida de servicios de tren que empiezan en la estación  $k$  y sea  $E_{end}^k \subset E_{train}$  el conjunto de los últimos eventos de llegada de servicios de tren que terminan su recorrido en la estación  $k$ . Además sea  $e_{rol}^k \in E_{inv}$  el inventario inicial de material rodante en la estación  $k$ .

Para cada estación  $k$ :

- Para un comienzo de un tren  $e \in E_{start}^k$  desde la estación  $k$ :
  - El conjunto  $A_{rol}^-(e)$  está formado por actividades  $a = (f, e)$  desde:
    1. Arribos correspondientes a trenes que terminan en la estación:  $f \in E_{end}^k$ .
    2. El evento de inventario  $e_{rol}^k$ .
  - El conjunto  $A_{rol}^+(e)$  es vacío, dado que el próximo evento en utilizar el material rodante es el arribo del tren  $t_e$  en la próxima estación. El traspaso se modela de manera implícita.

- Para un arribo final de un tren  $e \in E_{end}^k$  a la estación  $k$ :
  - El conjunto  $A_{rol}^-(e)$  es vacío por el mismo motivo mencionado arriba.
  - El conjunto  $A_{rol}^+(e)$  está formado por actividades  $a = (e, f)$  hacia:
    1. Salidas correspondientes a trenes que comienzan en la estación:  $f \in E_{start}^k$ .
- Para el evento de inventario relacionado a la estación  $k$ , el conjunto  $A_{rol}^+(e_{rol}^k)$  está formado por actividades  $a = (e_{rol}^k, e)$  hacia comienzos desde la estación  $k$ :  $e \in E_{start}^k$ .

Observar que efectivamente no hay actividades de material rodante en eventos intermedios del servicio de tren ya que no permitimos cambios de material rodante durante el recorrido. Sin embargo en el caso de una red geográficamente muy grande, puede ser necesario realizar cortes y entonces podría haber actividades de material rodante en eventos intermedios del servicio, pero este caso queda fuera del alcance de este trabajo.

La Figura 2.6 muestra el grafo de eventos y actividades del recurso de material rodante para la red de ejemplo 2.1 y el cronograma asociado 2.1. Para facilitar el entendimiento, se incluyen también las actividades de tren en líneas punteadas. Al igual que para los recursos anteriores, si bien por la definición de los conjuntos  $A_{rol}^-$  y  $A_{rol}^+$  el modelo permite que un evento traspase el recurso a otro evento programado en un tiempo anterior, dichas actividades no son incluidas dado que no podrían ser seleccionadas por las desigualdades (2.8). Para los eventos de inventario, se incluye entre paréntesis su respectiva capacidad. En este caso, los mismos representan las unidades disponibles en los patios de maniobras de las estaciones  $K$  y  $M$ , por lo que en este ejemplo tenemos 1 y 2 unidades respectivamente.

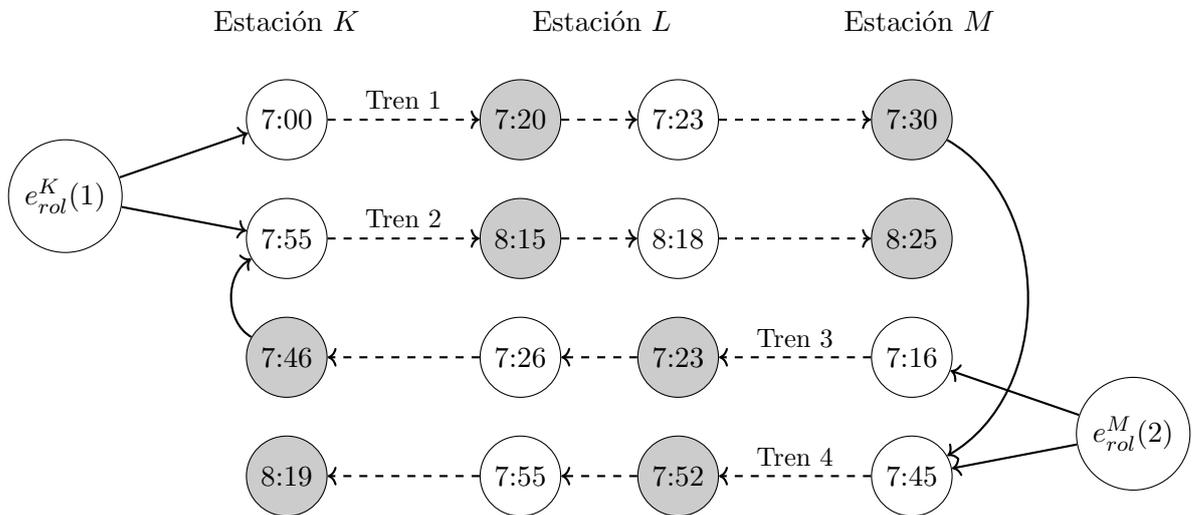


Fig. 2.6: Eventos y actividades relacionadas con el recurso de material rodante del cronograma ejemplo 2.1. Los eventos se organizan horizontalmente según su ubicación geográfica.

## 2.5. Bloqueo de una sección de vías

En la siguiente sección explicaremos las extensiones que deben realizarse sobre el modelo para contemplar correctamente el bloqueo de vías dentro de nuestro problema.

Recordemos, tal como introdujimos en la Sección 2.1.3, que el bloqueo afecta un tramo de vías entre dos estaciones. El mismo puede afectar solo algunas de las vías del tramo, en cuyo caso se tratará de un bloqueo parcial, o afectar todas las vías del tramo, produciendo un bloqueo total. En la misma sección introductoria comentamos que existen tres instantes de tiempo que le dan entidad al bloqueo. Por un lado tenemos el tiempo de inicio, al cual llamaremos en nuestro modelo  $\tau_1$ . Al tiempo de fin del bloqueo, que determina cuando las vías afectadas vuelven a estar disponibles, lo notaremos como  $\tau_2$ . Finalmente, el tiempo límite de transición desde el nuevo cronograma generado hacia el cronograma original lo notaremos  $\tau_3$ .

### 2.5.1. Trenes en el área del bloqueo

Los servicios de tren que pasen por el área del bloqueo deben ser tratados con especial atención. Supongamos que el bloqueo se produce en la sección de vías  $(s_1^B, s_2^B)$ . Sea  $T^{\text{disr}} \subset T$  el conjunto de los servicios de tren  $t$  tales que tienen programado un evento de salida de la estación  $s_1^B$  (en el cronograma original) dentro de la ventana de bloqueo (entre  $\tau_1$  y  $\tau_2$ ). Cada uno de los eventos de  $t$  se pueden clasificar como **stopping** o **pass-through**, en base a si el tren se detiene en la estación asociada al evento o pasa por la misma sin detenerse. Entonces sea  $k$  la última estación en la que se detiene  $t$  antes de  $s_1^B$  y sea  $l$  la primer estación en la que se detiene  $t$  luego de  $s_2^B$ , particionaremos los trenes  $t \in T^{\text{disr}}$  en los servicios parciales  $\alpha_t$ ,  $\beta_t$  y  $\gamma_t$ , delimitados por las estaciones  $k$  y  $l$ .

Por ejemplo, en la Figura 2.7 se observan dos ejemplos de particiones posibles, en un recorte de la red alrededor del bloqueo entre las estaciones Belgrano y Núñez. El Tren 1 se detiene en todas las estaciones, entonces podemos tomar  $\beta_t$  únicamente como el tramo del bloqueo. En cambio el Tren 2 no se detiene en Belgrano, con lo cual  $\beta_t$  debe comenzar en la última estación donde el tren se detiene antes de Belgrano, que en este caso es Lisandro de la Torre.

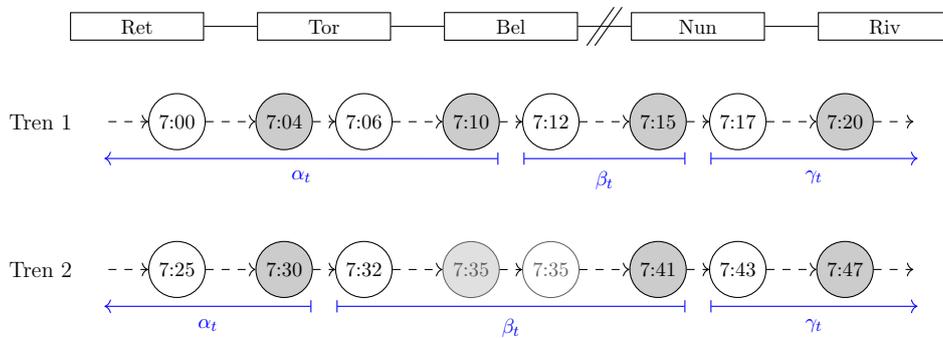


Fig. 2.7: Dos ejemplos de partición del servicio en el área del bloqueo. El bloqueo se produce entre las estaciones Belgrano ( $s_1^B$ ) y Núñez ( $s_2^B$ ). El Tren 1 se detiene en todas las estaciones, entonces  $k = s_1^B$  y  $l = s_2^B$ . En cambio el Tren 2 no se detiene en la estación Belgrano, con lo cual  $k$  se toma como la última estación en la que se detiene antes del bloqueo, que es Lisandro de la Torre.

Sean  $e^{\text{start}}$  y  $e^{\text{end}}$  el primer y último evento de  $t$  respectivamente. Sean  $e_k^{\text{arr}}$  y  $e_l^{\text{arr}}$  los eventos de arribo en las estaciones  $k$  y  $l$  respectivamente y sean  $e_k^{\text{dep}}$  y  $e_l^{\text{dep}}$  los eventos de salida. El servicio parcial  $\alpha_t$  realiza los eventos desde  $e^{\text{start}}$  hasta  $e_k^{\text{arr}}$ ,  $\beta_t$  realiza desde  $e_k^{\text{dep}}$  hasta  $e_l^{\text{arr}}$  y  $\gamma_t$  desde  $e_l^{\text{dep}}$  hasta  $e^{\text{end}}$ .

Esta forma de particionar los trenes nos permitirá cancelarlos parcialmente en vez de cancelar la totalidad del servicio en caso de no poder operar. Más aún, incluiremos las restricciones (2.10) y (2.11) para asegurar que si la sección  $\beta_t$  del servicio es operada ( $y_{\beta_t} = 0$ ), entonces tanto  $\alpha_t$  como  $\gamma_t$  serán operadas también. Esto es equivalente a operar el tren  $t$  completo.

$$y_{\beta_t} \geq y_{\alpha_t} \quad \forall t \in T^{\text{disr}} \quad (2.10)$$

$$y_{\beta_t} \geq y_{\gamma_t} \quad \forall t \in T^{\text{disr}} \quad (2.11)$$

Las actividades de recursos relacionadas con los eventos de los trenes parciales son exactamente las mismas que para los eventos del tren original  $t$ . Sin embargo, para asegurarnos que  $\alpha_t$ ,  $\beta_t$  y  $\gamma_t$  usen el mismo material rodante en caso de que ninguno sea cancelado, *agregamos* una actividad de material rodante desde el evento  $e_k^{\text{arr}}$  del tren  $\alpha_t$  hacia el evento  $e_k^{\text{dep}}$  del tren  $\beta_t$  y otra desde el evento  $e_l^{\text{arr}}$  de  $\beta_t$  hacia el evento  $e_l^{\text{dep}}$  del tren  $\gamma_t$ , ambas con tiempos de seguridad nulo. Además, estas son las únicas actividades de material rodante definidas para el tren  $\beta_t$ .

Por último, para el evento  $e_l^{\text{arr}}$ , en el caso de las actividades de material rodante, la desigualdad (2.6) se vuelve una igualdad, i.e., si  $\beta_t$  no se cancela, necesariamente le traspasa el material rodante a  $\gamma_t$ .

Con estos pequeños ajustes, aseguramos que si ninguno de los trenes parciales se cancela, todos usen el mismo material rodante. Por otro lado, en caso de que  $\beta_t$  sea cancelado, el material rodante usado por  $\alpha_t$  y  $\gamma_t$  puede ser utilizado por otros trenes desde las estaciones  $k$  y  $l$  respectivamente.

### 2.5.2. Eventos incluidos en el modelo

En el modelo presentado en las secciones anteriores incluimos todos los eventos de todos los servicios de tren del cronograma original. Pero si tenemos en cuenta que los eventos anteriores a  $\tau_1$  ya ocurrieron, podemos omitir muchos de los eventos con  $q_e < \tau_1$ . Sin embargo no podemos omitirlos a todos ya que si no estaríamos asumiendo que todos los recursos se encuentran disponibles en  $\tau_1$ , cuando en realidad podrían estar siendo usados, o bien ya fueron liberados pero debemos garantizar el tiempo de seguridad. Sin entrar en los detalles, se define un límite inferior de tiempo en base a los tiempos de seguridad de los distintos recursos, y todos los eventos posteriores a ese límite deberán ser incluidos para modelar correctamente el uso de recursos.

Por otro lado vimos que a partir de  $\tau_3$  los eventos ya no pueden ser demorados (para volver al cronograma original), entonces también podemos omitir muchos de los eventos con  $q_e > \tau_3$ . De forma análoga definiremos un límite superior de tiempo posterior a  $\tau_3$ , antes del cual todos los eventos deberán ser incluidos.

Además de asegurar el correcto uso de los recursos, debemos garantizar cierta cantidad de material rodante en cada estación al final del día. Para esto agregaremos eventos de inventario de **guardado de material rodante** en cada estación, cuya capacidad indicará la cantidad de composiciones que deben terminar allí dentro de los límites de tiempo mencionados, asumiendo que luego de  $\tau_3$  el cronograma continúa como el original y dicho material rodante terminará en las estaciones requeridas.

Como mencionamos al principio de este capítulo, para más detalles de la definición de los límites de tiempo y consideraciones especiales en los bordes, referirse a Mosteiro [2].

## 3. Modelo integrado de reasignación de tripulación

En el capítulo anterior estudiamos el problema de reprogramación de cronograma ferroviario y cómo reasignar los recursos de vías y material rodante. En el presente capítulo estudiaremos el problema de asignación del recurso de tripulación, necesario también para operar los servicios de tren y presentaremos un modelo PLE integrado.

### 3.1. Problema de reasignación de tripulación

Se pueden encontrar descripciones detalladas del problema de reasignación de tripulación en Veelenturf et al. [6], Potthoff y Huisman [4], Potthoff [5] y Huisman [7]. A continuación realizaremos una compilación de la información recolectada de dichas fuentes, introduciendo la nomenclatura del dominio, las restricciones y los objetivos.

La tripulación de un servicio de tren incluye a la persona que conduce el tren y podría incluir también guardas, mozos/as, etc. En Veelenturf et al. [6] se menciona que en el contexto de manejo de grandes interrupciones, generalmente se entiende a la tripulación como la persona que conduce el tren, es decir un recurso necesario e indivisible para operar un servicio de tren. El resto de la tripulación no es necesaria para que el tren cumpla con su cronograma y es por eso que no se tiene en cuenta para resolver este tipo de problemas.

Cada tripulación al inicio del día tiene asignado un plan o **jornada laboral** donde se especifican que tarea/s debe realizar durante el día. La jornada puede ser **activa**, i.e. operando servicios de tren, o **de reserva**, en el caso de que la tripulación permanece en una estación durante cierto período de tiempo, disponible para operar. Las jornadas empiezan y terminan en la misma estación, la cual llamaremos **base de la tripulación**.

Los servicios de tren se particionan en **tareas**, delimitadas por **estaciones de relevo**, que son todas aquellas que permiten a una tripulación pasar físicamente de un servicio de tren a otro. Este traspaso requiere de un tiempo mínimo para ocurrir, al que llamaremos **tiempo de conexión**. Observar que si la tripulación realiza dos tareas consecutivas en el mismo material rodante, el tiempo de conexión es nulo, ya que no hay un traspaso físico.

Las tareas que se le asignan a una tripulación en su jornada laboral pueden ser de dos tipos: de **conducción**, si la tripulación debe operar una tarea de un servicio de tren, o de **reposicionamiento**, si la tripulación debe viajar entre dos estaciones. Las tareas de reposicionamiento pueden realizarse enviando a la tripulación como **pasajera** de una tarea de un servicio de tren de la misma red, o bien puede realizarse por fuera, utilizando otro medio de transporte u otra red ferroviaria. Por simplicidad, asumiremos que los viajes por fuera de la red se realizan en **taxi**.

Ante un bloqueo inesperado de vías puede suceder que las jornadas originales no puedan completarse y se deban programar jornadas alternativas. Una **jornada alternativa** de trabajo consiste de las tareas originales hasta el momento del bloqueo y una compleción factible que permita a la tripulación terminar su jornada regresando a la base.

En la programación de jornadas alternativas, se debe tener en cuenta que cada

tripulación posee un **horario de inicio** y **horario de fin** de jornada, que es el tiempo que la tripulación está disponible para trabajar. Por ende la jornada alternativa no debería terminar después del horario de fin. Además debe contemplarse un tiempo de **almuerzo** o recreo que debe durar un mínimo de tiempo y realizarse aproximadamente en la mitad de la jornada. Por ejemplo un almuerzo de por lo menos 30 minutos y el trabajo realizado antes y después del almuerzo no puede exceder las 5 horas. El almuerzo sólo puede realizarse en ciertas estaciones que tienen las instalaciones necesarias, que por simplicidad asumiremos que todas las estaciones de relevo las tienen.

Como se menciona en Potthoff y Huisman [4], Potthoff [5] y brevemente en Dollevoet et al. [3], la reasignación de tripulación tiene varios objetivos. Por un lado es importante lograr la factibilidad, para minimizar las cancelaciones y demoras que impactan directamente a los usuarios. Pero por otro lado, hay costos operacionales que minimizar, como por ejemplo la realización de tareas de reposicionamiento, que implican pagar a la tripulación por horas que no son de manejo ni recreo, e incluso pagar tal vez un medio de transporte externo. Por último, dado que el problema se resuelve en tiempo real y que los cambios en las tareas asignadas los comunica un operador, realizar demasiados cambios a las tareas puede no ser viable e incluso llevar a errores de comunicación de los cambios que generen un resultado aún peor. Estos tres aspectos deben ser balanceados a la hora de realizar la reasignación.

Por ejemplo, en la red de la Figura 2.1 con los horarios de la Tabla 2.1, supongamos que las estaciones  $K$  y  $M$  son de relevo. En tal caso, cada uno de los servicios de tren consiste de una única tarea que corresponde a todo el recorrido, ya que no hay estaciones de relevo intermedias. Sea  $c_1$  una tripulación cuya estación base es  $K$  y supongamos que su horario de trabajo es desde las 7:00 hasta las 13:00, y sea  $c_2$  cuya base es  $M$  con horario de 7:00 a 13:00 también. Como se observa en la Figura 3.1, la asignación más simple en este caso es que  $c_1$  realice los trenes 1 y 4 completos, y que  $c_2$  realice los trenes 3 y 2 completos. Observar que el horario de almuerzo queda fuera de este ejemplo y debería ser posterior a estas tareas. En general para visualizar una jornada laboral, el formato de la Figura 3.1 resulta confuso, es por eso que utilizaremos un Gantt como en la Figura 3.2.

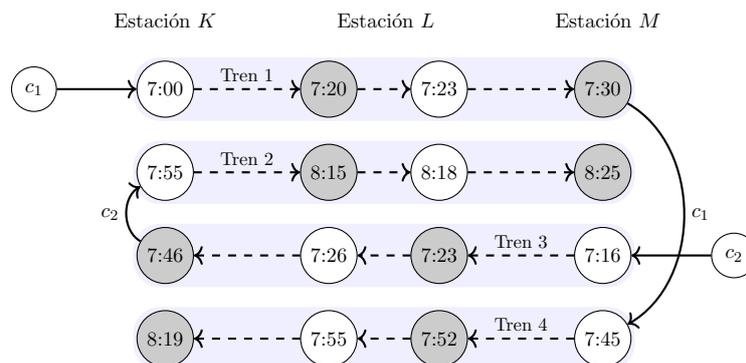


Fig. 3.1: Ejemplo a modo ilustrativo de una asignación de tripulación a tareas de manejo. Las tareas agrupan los eventos de salidas y arribos de cada tren en el área azul.

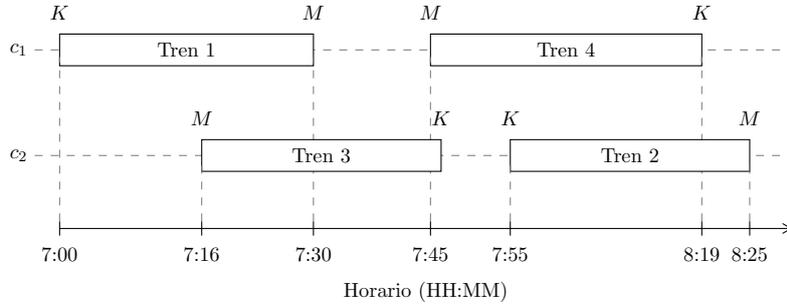


Fig. 3.2: Gantt del ejemplo de la Figura 3.1. Las tareas de manejo son representadas con cajas blancas, las cuales indican el número de tren en el interior y las estaciones de origen y destino en los extremos superiores. En el eje vertical se encuentran las asignaciones de tareas a cada tripulación (jornada laboral) y en el eje horizontal el tiempo.

### 3.2. Modelo matemático

En esta sección presentaremos una base del modelo matemático para el problema de reasignación de tripulación y definiremos el grafo de alcance (similar al definido en Potthoff [5]), que nos permitirá explorar el espacio de soluciones de una forma más simple.

Sea  $S^{rel} \subset S$  el subconjunto de las estaciones de relevo, es decir aquellas estaciones que permiten que una tripulación pase de un tren a otro (con distinto material rodante). Sea  $T$  el conjunto de los servicios de tren en el cronograma original. Cada tren  $t \in T$  se puede particionar en tareas  $\theta$  que empiezan y terminan en cada una de las estaciones de relevo donde el tren se detenga (i.e. no particionamos en una estación de relevo por la que el tren pasa sin detenerse). Notar que esta partición es distinta a la que realizamos previamente en la Sección 2.5.1 en  $\alpha_t, \beta_t, \gamma_t$ , esta nueva partición es independiente del bloqueo y está orientada a dividir cada servicio en partes que pueden ser asignadas independientemente a las tripulaciones. Asumiremos que las estaciones donde cualquier servicio empieza o termina es de relevo. De esta manera, tendremos  $t = [\theta_1, \dots, \theta_n]$ . Sea entonces  $\Theta$  el conjunto de todas las tareas de todos los trenes en  $T$ .

Cada tarea  $\theta \in \Theta$  abstrae una secuencia de eventos de arribo y salida del tren correspondiente, al que nos referiremos como  $t_\theta$ . Los dos eventos más importantes son la primera salida y el último arribo, ya que delimitan a la tarea y determinan el tiempo de inicio y de fin. Entonces sea  $\theta = [[e, f]]$  con  $e, f \in E_{train}$  la primer salida y el último arribo respectivamente. Observar que son eventos de tren, es decir que  $q_e, q_f$  son los tiempos en el cronograma original,  $d_e, d_f$  son las máximas demoras de cada uno y  $x_e, x_f$  son las variables que indican los tiempos en el nuevo cronograma, del modelo del Capítulo 2.

Por ejemplo supongamos que tenemos una red con sólo cinco estaciones como se muestra en la Figura 3.3 y el servicio de tren 3001 ( $t$ ) opera desde Retiro a Rivadavia, partiendo a las 7:00. Las estaciones de inicio y fin del tren son de relevo. La primera tarea  $\theta_1$  va desde Retiro hasta Belgrano, que es la primera estación de relevo en la que el tren se detiene (se detiene antes en Lisandro de la Torre, que no es una estación de relevo). Luego la segunda tarea parte de Belgrano y llega hasta Rivadavia, ya que a pesar de que Núñez es de relevo, el servicio 3001 no se detiene allí. Luego  $t = [\theta_1, \theta_2]$ .

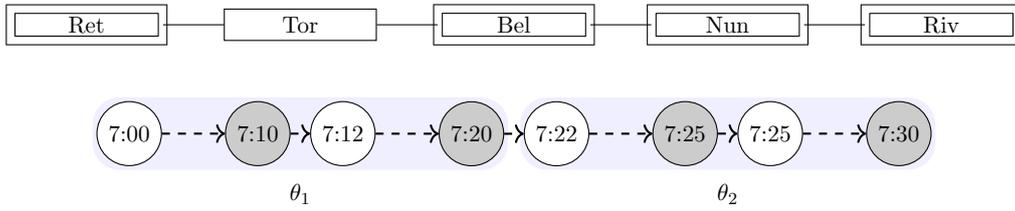


Fig. 3.3: Ejemplo de una partición de un tren en tareas. El eje horizontal representa la distribución geográfica de las estaciones, las cuales se encuentran en la fila superior. Las estaciones con línea doble son de relev  $S^{rel}$ . En la fila inferior se encuentran los eventos del servicio de tren 3001, los blancos son las salidas y los grises son los arribos. El horario corresponde a la programación original, i.e.  $q_e$ . En la estación Núñez el tren no se detiene. Las dos tareas  $\theta_1, \theta_2$  resultantes de la partición se encuentran sombreadas en azul.

### 3.2.1. Tiempos de conexión

Dadas dos tareas  $\theta_i, \theta_j \in \Theta$ , llamamos **tiempo de conexión** al tiempo mínimo que necesita cualquier tripulación entre que finaliza  $\theta_i$  y puede empezar  $\theta_j$ . Como mencionamos anteriormente, este tiempo sirve para que la tripulación pase de un material rodante a otro, con lo cual si el material rodante es el mismo, el tiempo de conexión es nulo.

Como vimos en el Capítulo 2, cuando reprogramamos el cronograma ferroviario incorporando restricciones de material rodante, limitamos el modelo para que un servicio de tren no pueda cambiar su material rodante. Esta suposición tiene sentido práctico en el transporte de pasajeros ya que típicamente un mismo servicio usará siempre la misma composición de material rodante.

Siendo nuestro objetivo en este trabajo integrar restricciones de tripulación al modelo del Capítulo 2, podemos asumir que dos tareas utilizan el mismo material rodante si corresponden al mismo servicio de tren. Por simplicidad, para el resto de las conexiones tomaremos un tiempo fijo idéntico para todas, aunque tomar un valor distinto en cada conexión no agregaría complejidad al problema. Luego definimos el tiempo de conexión de la siguiente manera:

$$\text{conn}(\theta_i, \theta_j) = \begin{cases} 0 & \text{si } t_{\theta_i} = t_{\theta_j} \\ \mathbb{P}_{\text{conn}} & \text{cc.} \end{cases} \quad (3.1)$$

siendo  $\mathbb{P}_{\text{conn}} > 0$  un parámetro del problema. Típicamente  $\mathbb{P}_{\text{conn}}$  será como máximo de 5 minutos aproximadamente. Observar que una tripulación podría realizar dos tareas consecutivas de diferentes trenes, pero en el mismo material rodante. En este caso asumimos que es necesario un tiempo de conexión no nulo, por ejemplo para un descanso de la tripulación entre un servicio y otro.

### 3.2.2. Grafo de alcance

En el momento que ocurre un bloqueo, las jornadas originales de las tripulaciones se cumplieron hasta cierto instante de tiempo. A partir de ahí, hay muchas formas de completar la jornada alternativa, adaptada al bloqueo.

En Potthoff [5] (secciones 3.4.2 y 4.3.4), para cada jornada original  $\delta$ , se construye un grafo  $G^\delta$  que modela cada una de estas alternativas como caminos, y se utiliza un algoritmo de camino mínimo para seleccionar la más conveniente y alimentar un algoritmo de generación de columnas. En nuestro caso, sólo estamos interesados en la estructura del mismo, no en los pesos ya que el mejor camino será determinado en forma global teniendo en cuenta todas las tripulaciones y los rangos horarios de los eventos de inicio y fin de las tareas.

De esta manera definimos el **grafo de alcance**, un grafo dirigido que codifica posibles secuencias de tareas. Los vértices del grafo representan tareas y un arco

entre dos tareas  $\theta_i \rightarrow \theta_j$  indica que una tripulación que realiza la tarea  $\theta_i$  podría realizar la tarea  $\theta_j$  inmediatamente después.

Observar que a diferencia de Potthoff [5], este grafo no se construye para una jornada de trabajo original (i.e. una tripulación específica), sino que se arma en base a  $\Theta$ , es decir las tareas de todos los trenes.

Sea entonces  $G^R(\Theta) = (\Theta, R)$  un grafo dirigido con  $\Theta$  el conjunto de nodos y  $R$  el conjunto de aristas. El conjunto  $R$  define la **relación de alcance**, donde  $\theta_i \rightarrow \theta_j$  si y sólo si:

- (I) La estación de finalización de  $\theta_i$  debe ser igual a la estación de inicio de  $\theta_j$ .
- (II) La tarea  $\theta_i$  puede finalizar antes de que pueda empezar  $\theta_j$ , con el suficiente tiempo de conexión entre ellas.

Para analizar cada condición, supongamos que  $\theta_i = [[e_i, f_i]]$  y  $\theta_j = [[e_j, f_j]]$ . La primera condición es la más simple, alcanza con pedir que  $f_i \in E_{arr}^k$  y  $e_j \in E_{dep}^k$  para la misma estación  $k \in S$ .

La segunda condición es más complicada, ya que los tiempos de inicio y finalización de cada tarea son variables en nuestro problema. En esencia debemos pedir:

$$x_{f_i} + \text{conn}(\theta_i, \theta_j) \leq x_{e_j}$$

i.e. que el tiempo de inicio de  $\theta_j$  sea por lo menos el tiempo de finalización de  $\theta_i$  más el tiempo de conexión. Recordemos que el objetivo del grafo es codificar las posibles secuencias de tareas, entonces nos gustaría agregar  $\theta_i \rightarrow \theta_j$  si en alguno de los posibles valores de las variables, la desigualdad se cumple, y luego asegurar su cumplimiento en el modelo PLE. Por las restricciones (2.2) y (2.3) del modelo PLE del Capítulo 2, tenemos que  $x_e \in [q_e, q_e + d_e]$  para cualquier evento  $e$ . Acotando ambos lados de la desigualdad:

$$q_{f_i} + \text{conn}(\theta_i, \theta_j) \leq x_{f_i} + \text{conn}(\theta_i, \theta_j) \leq x_{e_j} \leq q_{e_j} + d_{e_j}$$

En la Figura 3.4 se ilustra esta segunda condición.

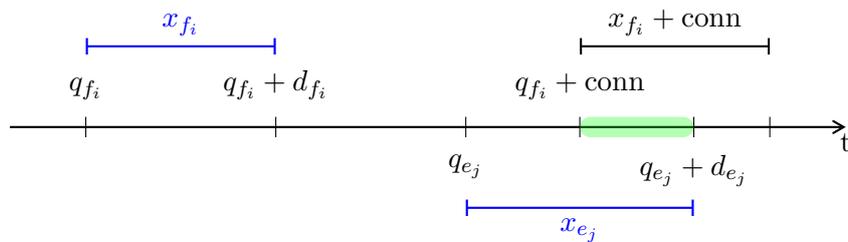


Fig. 3.4: Ejemplo de la condición (3.3) del grafo de alcance. Sean  $\theta_i = [[e_i, f_i]]$  y  $\theta_j = [[e_j, f_j]]$  dos tareas del modelo tales que  $f_i$  y  $e_j$  son eventos de la misma estación. En el eje horizontal se representa el tiempo y en azul se observan los intervalos de tiempo en los que pueden ocurrir los eventos  $f_i$  y  $e_j$  en la reprogramación. Si no pudiéramos demorar los eventos, es decir si sólo tenemos en cuenta los horarios del cronograma original para armar  $R$ , luego  $(\theta_i \rightarrow \theta_j) \notin R$ , ya que  $q_{f_i} + \text{conn} > q_{e_j}$ . Sin embargo si tenemos en cuenta que los eventos pueden ser demorados en el cronograma alternativo, entonces  $(\theta_i \rightarrow \theta_j) \in R$  ya que se cumple (3.3) (sombreado en verde).

Formalmente, siendo  $\theta_i = [[e_i, f_i]]$  y  $\theta_j = [[e_j, f_j]]$ , definimos

$$(\theta_i \rightarrow \theta_j) \in R \quad \Leftrightarrow \quad f_i \in E_{arr}^k, e_j \in E_{dep}^k, k \in S \quad (3.2)$$

$$q_{f_i} + \text{conn}(\theta_i, \theta_j) \leq q_{e_j} + d_{e_j} \quad (3.3)$$

Analicemos las propiedades del grafo de alcance  $G^R(\Theta)$ .

**Prop. 3.1.** Si  $d_e = 0$  para todo  $e$ , luego el grafo es acíclico.

*Demostración.* Si los delays son nulos, la desigualdad (3.3) requiere que las tareas se conecten según sus horarios originales y el tiempo de conexión, con lo cual los arcos *avanzan* con el tiempo (el origen tiene un tiempo de ocurrencia menor o igual al del destino). Si hubiera un ciclo, significaría que por un lado  $\theta_i$  finaliza antes del comienzo de  $\theta_j$ , pero al mismo tiempo  $\theta_j$  finaliza antes del comienzo de  $\theta_i$ , lo cual no es posible.

Más formalmente, para que haya un ciclo entre  $\theta_i = [[e_i, f_i]]$  y  $\theta_j = [[e_j, f_j]]$ , estas dos desigualdades se deben cumplir:

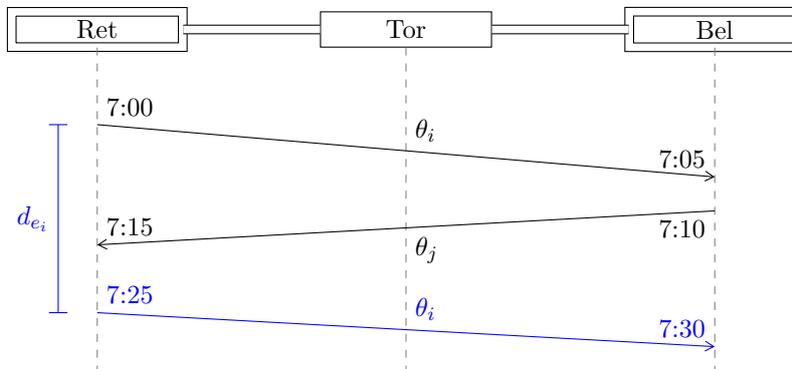
$$(a) \quad q_{f_i} + \text{conn}(\theta_i, \theta_j) \leq q_{e_j} \quad \wedge \quad (b) \quad q_{f_j} + \text{conn}(\theta_j, \theta_i) \leq q_{e_i}$$

Partiendo de (b) y teniendo en cuenta que las tareas tienen una duración no nula (i.e.  $q_{e_i} < q_{f_i}$  y  $q_{e_j} < q_{f_j}$ ):

$$\underbrace{q_{f_i} + \text{conn}(\theta_i, \theta_j)}_{(a)} + \underbrace{\text{conn}(\theta_j, \theta_i)}_{q_{e_j} < q_{f_j}} < \underbrace{q_{f_j} + \text{conn}(\theta_j, \theta_i)}_{(b)} \leq q_{e_i}$$

Observar que llegamos a que  $q_{f_i} < q_{e_i}$  ya que los tiempos de conexión son no negativos. Pero esto es absurdo porque significa que la tarea  $\theta_i$  finaliza antes de comenzar. Esta idea se puede extender inductivamente a ciclos de cualquier longitud.  $\square$

En el caso general, el grafo puede tener ciclos por ejemplo si una tarea  $\theta_i$  puede ser realizada antes que  $\theta_j$  en el cronograma original, y las demoras máximas permitidas en la reprogramación son suficientemente grandes para que  $\theta_j$  pueda hacerse antes de  $\theta_i$ . Además debe cumplirse que ambas tareas unan el mismo par de estaciones. En la Figura 3.5 se observa un ejemplo de este caso.



*Fig. 3.5:* Ejemplo donde el grafo de alcance tiene un ciclo. El comienzo de la tarea  $\theta_i = [[e_i, f_i]]$  puede ser demorado como máximo  $d_{e_i} = 25$  minutos. Sea  $\theta_j = [[e_j, f_j]]$  y supongamos que el tiempo mínimo de conexión es de 5 minutos. Luego, por un lado tenemos que  $q_{f_i} + 5 \leq q_{e_j} + d_{e_j}$  y entonces  $(\theta_i \rightarrow \theta_j) \in R$ . Por ejemplo si  $f_i$  y  $e_j$  no son demorados (i.e. se realizan en horarios 7:05 y 7:10 resp.), se puede realizar la conexión y por ende realizar  $\theta_j$  después de  $\theta_i$ . Por otro lado tenemos que  $q_{f_j} + 5 \leq q_{e_i} + d_{e_i}$ , entonces  $(\theta_j \rightarrow \theta_i) \in R$ . Por ejemplo si  $f_j$  no se demora y  $e_i$  se demora el máximo posible (i.e. se realizan en horarios 7:15 y 7:25 resp.), se puede realizar la conexión y por ende realizar  $\theta_i$  después de  $\theta_j$ .

En la Figura 3.6 se presenta un ejemplo de cronograma y partición en tareas, y en la Figura 3.7 el grafo de alcance asociado. Observar que  $\theta_1$  no tiene ningún arco entrante. Es decir que para asignar tripulación a esa tarea, no puede tomarse de otra tarea, alguna tripulación debe encontrarse en la estación donde empieza  $\theta_1$  antes del tiempo de inicio para que pueda ser asignada a la misma.

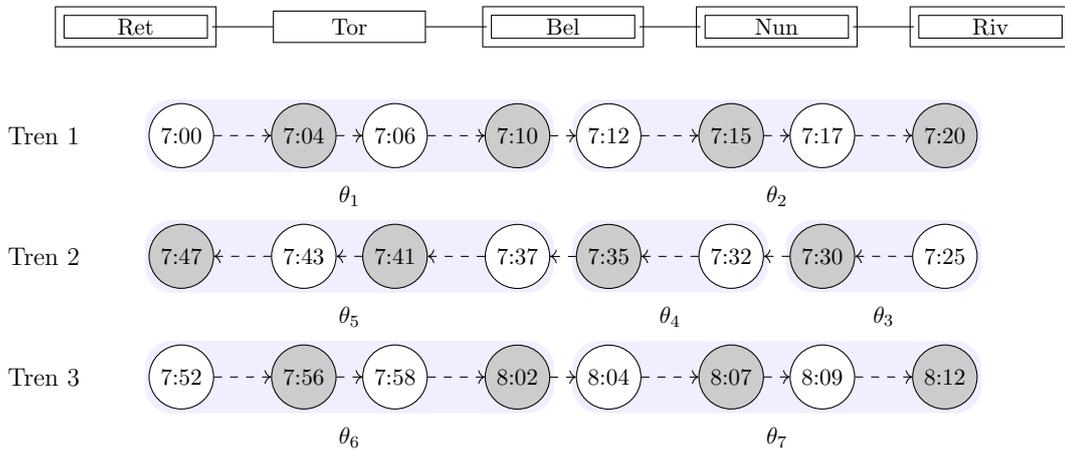


Fig. 3.6: Ejemplo de un cronograma de tres trenes entre Retiro y Rivadavia. El eje horizontal representa la distribución geográfica de las estaciones. El eje vertical agrupa los eventos por su servicio de tren correspondiente. Las estaciones con caja de doble línea son estaciones de relevo. Los nodos redondos son eventos de tren, los blancos son las salidas y los grises los arribos. La partición de cada tren en tareas se representa con el fondo sombreado de color y debajo se indica el nombre de la tarea.

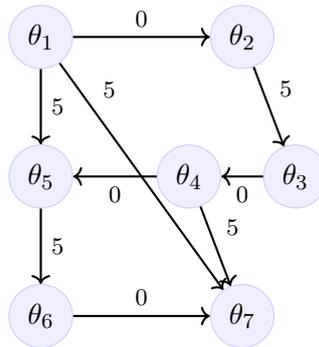


Fig. 3.7: Ejemplo de un grafo de alcance armado en base a los horarios y tareas definidos en la Figura 3.6, respetando la organización espacial de las tareas. Cada arista del grafo indica el tiempo mínimo de conexión  $\text{conn}(\theta_i, \theta_j)$  entre las tareas que une, siendo 0 para tareas del mismo servicio o de 5 para tareas de distintos servicios. Ningún evento es demorable ( $d_e = 0$  para todo  $e \in E_{train}$ ).

### 3.2.3. Red de flujo de tripulación

A continuación veremos cómo utilizar el grafo de alcance  $G^R(\Theta)$  para armar una red que codifique las posibles jornadas alternativas de cada tripulación.

Sea  $\mathbb{C}$  el conjunto de todas las tripulaciones. Para cada  $c \in \mathbb{C}$  construimos una **red de flujo**  $N_c(\Theta)$  a partir de un subgrafo de  $G^R(\Theta)$ . Además de las tareas, la red  $N_c(\Theta)$  tiene un **nodo inicial** o fuente del flujo  $e_{beg}^c$  y un **nodo final** o sumidero  $e_{end}^c$ . La capacidad del nodo fuente es  $i_{e_{beg}^c} = 1$  y el flujo asignado a cada arista debe ser entero, i.e. 0 o 1 (limitado por la capacidad). De esta manera, un flujo válido en la red determina una secuencia factible de tareas, es decir una jornada alternativa para  $c$ .

En esta red encontraremos tres tipos de aristas: las iniciales ( $e_{beg}^c, \theta$ ), las finales ( $\theta, e_{end}^c$ ) y las que transportan el flujo ( $\theta, \bar{\theta}$ ). Para definir las iniciales y finales, además de los horarios originales de las tareas y demoras permitidas, debemos tener en cuenta los horarios de disponibilidad de  $c$ , i.e. el **horario de inicio** de la jornada laboral  $w_c^{on}$  y el **horario de fin** de la jornada laboral  $w_c^{off}$ . También debemos tener en cuenta la **estación base**  $b_c$  de la tripulación, lugar donde debe comenzar y terminar su jornada.

Formalmente, sea  $N_c(\Theta) = (\{e_{beg}^c, e_{end}^c\} \cup \Theta, A_{cr}(c))$  la red de flujo de la tripulación  $c \in \mathbb{C}$ . Veamos como definir las aristas  $A_{cr}(c)$ . Para simplificar la notación, definiremos cada arista como  $a = (e, f)$  sin indicar explícitamente la tripulación asociada, que siempre quedará definida por el contexto. Para cada tarea  $\theta \in \Theta$ ,  $\theta = [[e, f]]$ , definimos las aristas de entrada y de salida:

- El conjunto  $A_{cr}^-(c, \theta)$  de aristas de entrada a  $\theta$  está formado por:
  - $a = (e_{beg}^c, \theta)$  si  $\theta$  sale de la estación base de  $c$  y puede hacerlo después de que  $c$  inicia su jornada laboral:

$$e \in E_{dep}^{bc} \quad \text{y} \quad w_c^{on} \leq q_e + d_e$$

- $a = (\bar{\theta}, \theta)$  si  $\theta$  puede realizarse después de  $\bar{\theta}$ , i.e.  $a \in G^R(\Theta)$ .

- El conjunto  $A_{cr}^+(c, \theta)$  de aristas de salida de  $\theta$  está formado por:
  - $a = (\theta, \bar{\theta})$  si  $\bar{\theta}$  puede realizarse después de  $\theta$ , i.e.  $a \in G^R(\Theta)$ .
  - $a = (\theta, e_{end}^c)$  si  $\theta$  termina en la base de  $c$  y puede hacerlo antes de que  $c$  termine su jornada laboral:

$$f \in E_{arr}^{bc} \quad \text{y} \quad q_f \leq w_c^{off}$$

Observar que con esta definición podría haber aristas  $(\theta, e_{end}^c)$  que no son alcanzables desde  $e_{beg}^c$ , por ejemplo si  $\theta$  es una tarea que empieza antes de  $w_c^{on}$ . En la práctica para eliminar estos casos alcanza con recorrer el grafo desde  $e_{beg}^c$  marcando los nodos alcanzables y luego eliminar todas las aristas  $(\theta, e_{end}^c)$  correspondientes a tareas no alcanzables. Por otro lado, podría suceder que  $e_{end}^c$  no fuera alcanzable desde  $e_{beg}^c$ , pero esto es un caso borde que analizaremos en secciones posteriores, ya sea no utilizando la tripulación en la solución del modelo PLE, o utilizándola parcialmente, por ejemplo terminando excepcionalmente la jornada en una estación que no sea su base o usando reposicionamiento en taxi.

Otro punto a considerar es que tanto las aristas iniciales como las finales podrían ser o no factibles según los horarios de reprogramación de  $\theta = [[e, f]]$ ,  $x_e$  y  $x_f$ . El modelo PLE deberá garantizar que  $w_c^{on} \leq x_e$  para tareas iniciales y  $x_f \leq w_c^{off}$  para tareas finales. En el grafo estamos incluyendo todas las que *podrían* ser factibles, teniendo en cuenta que  $x_e \in [q_e, q_e + d_e]$ . Este criterio es análogo al que utilizamos anteriormente para construir  $G^R(\Theta)$ .

En la Figura 3.8 se presentan dos ejemplos de las redes de flujo descritas, basadas en el ejemplo de la Figura 3.6.

Una vez construída la red  $N_c(\Theta)$  para cada  $c \in \mathbb{C}$ , vamos a unificar estas redes. Esto nos permitirá integrarnos más fácilmente con el modelo de Veulenturf et al. [1] descrito en el Capítulo 2.

Sea  $N_{\mathbb{C}}(\Theta) = (\mathbb{I}_{\mathbb{C}} \cup \mathbb{F}_{\mathbb{C}} \cup \Theta, A_{cr})$  la **red de flujo de tripulación**, tal que  $\mathbb{I}_{\mathbb{C}} = (e_{beg}^c)_{c \in \mathbb{C}}$  es el conjunto de los nodos iniciales,  $\mathbb{F}_{\mathbb{C}} = (e_{end}^c)_{c \in \mathbb{C}}$  el de los nodos finales y  $A_{cr} = \cup_{c \in \mathbb{C}} A_{cr}(c)$  la unión de las aristas de cada tripulación  $c$ .

Una forma de entender esta red es como una red de flujos de múltiples recursos o *commodities*. Cada tripulación es un recurso de esta red, con capacidad 1. Cada arista de la red sólo puede transportar un tipo de recurso, i.e. una tripulación específica  $c$ , y en una cantidad entera (0 o 1). Además cada nodo (cada tarea) admite una cantidad ilimitada de recursos (que más adelante limitaremos en el modelo PLE). Luego, un flujo en esta red representa una asignación de jornadas alternativas para cada  $c$ .

En la Figura 3.9 se presenta la red de flujo de tripulación para los ejemplos analizados en las Figuras 3.6, 3.7 y 3.8.

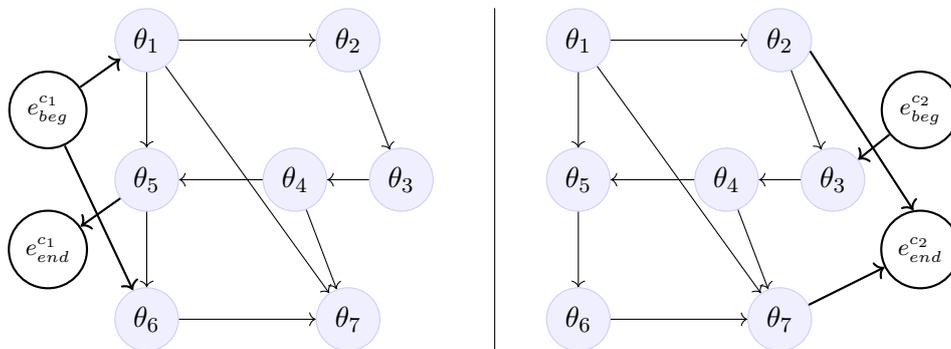


Fig. 3.8: Ejemplo de redes de flujo  $N_{c_1}(\Theta)$  y  $N_{c_2}(\Theta)$  (izquierda y derecha resp.) para el ejemplo de las Figuras 3.6 y 3.7, siendo que  $c_1$  tiene como base a Retiro y  $c_2$  Rivadavia, ambos con horario de 7:00 a 13:00. Por un lado,  $c_1$  puede empezar desde cualquiera de las tareas que salen de Retiro, ya que ambas son posteriores a su horario de inicio y puede terminar  $\theta_5$  que es la única que termina en su base. Observar que en este escenario, no es factible asignar a  $c_1$  a  $\theta_6$  o  $\theta_7$ , ya que luego no podría retornar a su base. Por otro lado,  $c_2$  puede empezar desde  $\theta_3$  que es la única tarea que sale de Rivadavia, lo que implica que  $\theta_1, \theta_2$  son inalcanzables para  $c_2$ , con lo cual podríamos eliminarlas de  $N_{c_2}(\Theta)$ .

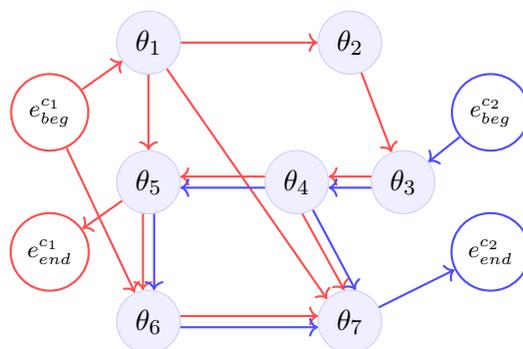


Fig. 3.9: Ejemplo de red de flujo de tripulación  $N_C(\Theta)$  generada a partir de las subredes de la Figura 3.8. Usamos los colores para distinguir las aristas correspondientes a cada tripulación. Observar que omitimos las aristas inalcanzables para  $c_2$ .

### 3.3. Red de Evento-Actividad Extendida

En la sección anterior presentamos la base del modelo matemático del problema de reasignación de tripulación. En esta sección estudiaremos cómo integrar este modelo al utilizado en el Capítulo 2 para la reprogramación de cronograma ferroviario.

Como mencionamos en secciones anteriores, la tripulación es un recurso necesario para operar servicios de tren. En la red de evento-actividad  $N = (E, A)$  del Capítulo 2 sólo tuvimos en cuenta las vías (de tramo y estación) y el material rodante. Para agregar el recurso de tripulación, por un lado debemos agregar los eventos de inventario  $E_{crew}$  y por otro las actividades de este recurso.

Los **eventos de inventario** de tripulación son los nodos que ya definimos como iniciales y finales, i.e.  $E_{crew} = \mathbb{I}_C \cup \mathbb{F}_C$ , cada uno con capacidad 1. La capacidad de los eventos iniciales modela el inventario del recurso al inicio del día y la de los eventos finales indica el inventario que debería haber al final del día.

Las **actividades de tripulación** se pueden mapear uno a uno con las aristas de la red de flujo de tripulación  $A_{cr}$ . Recordemos que una tarea  $\theta = [[e, f]]$  está definida como una abstracción de una secuencia de eventos de tren, desde una partida  $e$  hasta un arribo  $f$ . Luego, para cada tripulación  $c \in \mathbb{C}$  y cada arista  $a \in A_{cr}(c)$ , definimos el conjunto  $\bar{A}_{cr}$  de actividades de tripulación de la siguiente manera:

- Si  $a = (e_{beg}^c, \theta)$ , con  $\theta = [[e, f]]$ , luego  $(e_{beg}^c, e) \in \bar{A}_{cr}$ .

- Si  $a = (\theta, e_{end}^c)$ , con  $\theta = [[e, f]]$ , luego  $(f, e_{end}^c) \in \bar{A}_{cr}$ .
- Si  $a = (\theta, \theta)$ , con  $\theta = [[e, f]]$ ,  $\theta = [[\mathbf{e}, \mathbf{f}]]$ , luego  $\bar{a} = (f, \mathbf{e}) \in \bar{A}_{cr}$ , con  $L_{\bar{a}} = \text{conn}(\theta, \theta)$ .

Para simplificar la notación, usaremos  $A_{cr}$  indistintamente para referirnos a las aristas de  $N_C(\Theta)$  como para referirnos a las actividades de tripulación de la red de evento-actividad. Quedará claro del contexto cuál estamos usando por las convenciones de notación:  $\theta$  para tareas y  $e, f$  para eventos.

Incorporando estos eventos y actividades a la red  $N = (E, A)$  del Capítulo 2 obtenemos una nueva red de evento-actividad extendida que incluye el recurso de tripulación. Mantendremos la notación original de  $N = (E, A)$  para simplificar la lectura, pero de aquí en más siempre nos estaremos refiriendo a la red extendida.

En la Figura 3.10 presentamos un recorte de la red de evento-actividad extendida para el ejemplo de la Figura 3.6. Notar que la red  $N_C(\Theta)$  que armamos en la sección previa está *por encima* de la red de evento-actividad, es decir que abstrae detalles de la red  $N = (E, A)$  que no son necesarios para determinar una reasignación factible de tripulación. La utilidad de esto se verá cuando definamos matemáticamente las restricciones de tripulación.

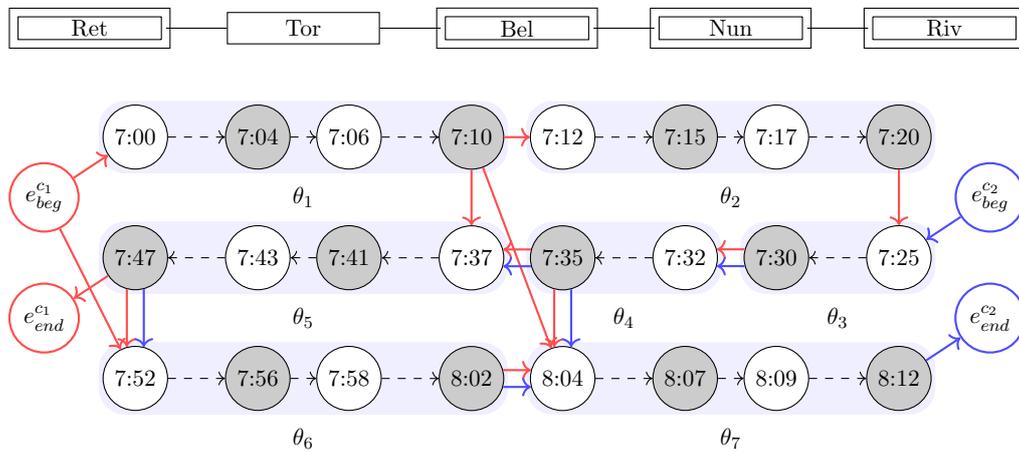


Fig. 3.10: Ejemplo de actividades de tripulación en una red de evento-actividad extendida correspondiente al ejemplo de la Figura 3.6 y a la red de flujo de tripulación de la Figura 3.9. Las líneas punteadas son las actividades de tren. Observar que omitimos las actividades de tren entre tareas del mismo tren para simplificar el gráfico.

### 3.4. Modelo PLE

Antes de definir las restricciones del recurso de tripulación, veamos qué impacto tiene agregar los eventos y actividades de inventario a  $E_{inv}$  y  $A_{inv}$  en el modelo del Capítulo 2. Los conjuntos  $\mathcal{A}^-(e) = \{A_{track}^-(e), A_{stat}^-(e), A_{rol}^-(e)\}$  y  $\mathcal{A}^+(e) = \{A_{track}^+(e), A_{stat}^+(e), A_{rol}^+(e)\}$  no son modificados, ya que las restricciones de flujo de tripulación se modelan aparte, sobre la red de tripulación  $N_C(\Theta)$ .

Observar que se agregan restricciones del tipo (2.8), que nos garantizarán que si una actividad  $a \in A_{cr}$  es utilizada por la solución ( $z_a = 1$ ), luego el tiempo mínimo de conexión de tripulación  $L_a$  se cumplirá.

También se agregan las variables  $z_a$  para  $a \in A_{cr}$  en las restricciones (2.9) con sus posibles valores.

Por último notar que a pesar de haber agregado eventos de inventario, no se agregan restricciones del tipo (2.7) ya que los eventos  $e \in E_{crew}$  tendrán  $C = \emptyset$  para todo  $C \in \mathcal{A}^+(e)$ , es decir no tendrán actividades de otros recursos.

### 3.4.1. Restricciones del recurso de tripulación

Al modelo de reprogramación de cronogramas, agregaremos las siguientes restricciones.

$$x_f - w_c^{on} + (1 - z_a)M_f^c \geq 0 \quad \forall c \in \mathbb{C}, a = (e_{beg}^c, f) \in A_{cr}^+(e_{beg}^c) \quad (3.4)$$

$$w_c^{off} - x_f + (1 - z_a)\bar{M}_f^c \geq 0 \quad \forall c \in \mathbb{C}, a = (f, e_{end}^c) \in A_{cr}^-(e_{end}^c) \quad (3.5)$$

$$\sum_{a \in A_{cr}^+(e_{beg}^c)} z_a \leq 1 \quad \forall c \in \mathbb{C} \quad (3.6)$$

$$\sum_{a \in A_{cr}^-(c, \theta)} z_a = \sum_{a \in A_{cr}^+(c, \theta)} z_a \quad \forall \theta \in \Theta, c \in \mathbb{C} \quad (3.7)$$

$$\sum_{a \in A_{cr}^+(e_{beg}^c)} z_a = \sum_{a \in A_{cr}^-(e_{end}^c)} z_a \quad \forall c \in \mathbb{C} \quad (3.8)$$

$$(1 - y_{t_\theta}) \leq \sum_{a \in A_{cr}^-(\theta)} z_a \quad \forall \theta \in \Theta \quad (3.9)$$

$$(1 - y_{t_\theta})(\mathbb{P}_{pas} + 1) \geq \sum_{a \in A_{cr}^-(\theta)} z_a \quad \forall \theta \in \Theta \quad (3.10)$$

$\mathbb{C}$	conjunto de tripulaciones
$w_c^{on}$	horario de inicio de $c$
$w_c^{off}$	horario de fin de $c$
$e_{beg}^c$	evento de inventario inicial de $c$
$e_{end}^c$	evento de inventario final de $c$
$A_{cr}^+(e)$	actividades de traspaso de cualquier tripulación desde el evento $e$
$A_{cr}^-(e)$	actividades de traspaso de cualquier tripulación al evento $e$
$A_{cr}^+(\theta)$	actividades de traspaso de cualquier tripulación desde la tarea $\theta$
$A_{cr}^-(\theta)$	actividades de traspaso de cualquier tripulación a la tarea $\theta$
$A_{cr}^+(c, \theta)$	actividades de traspaso de la tripulación $c$ desde la tarea $\theta$
$A_{cr}^-(c, \theta)$	actividades de traspaso de la tripulación $c$ a la tarea $\theta$
$\mathbb{P}_{pas}$	máxima cantidad de tripulaciones como pasajeras

Tab. 3.1: Notación formulación PLE con tripulación.

Dado que cada tripulación  $c$  trabaja sólo en una ventana de disponibilidad  $[w_c^{on}, w_c^{off}]$ , las restricciones (3.4) y (3.5) aseguran que los horarios de las tareas inicial y final de  $c$  respeten esta ventana. Específicamente, si se utiliza la actividad  $a = (e_{beg}^c, f)$ , i.e.  $z_a = 1$ , siendo  $f$  el primer evento de alguna tarea  $\theta = [[f, e]]$ , la restricción (3.4) correspondiente garantiza que  $x_f \geq w_c^{on}$ , es decir que la primera tarea que realiza  $c$  empieza después de  $w_c^{on}$ . Cuando  $z_a = 0$ , i.e. no sea utilizada en la solución, tomamos  $M_f^c \stackrel{\text{def}}{=} w_c^{on} - q_f$  para que la desigualdad se cumpla sin importar el valor de  $x_f$ . Notar que si  $w_c^{on} = q_f$ , la restricción es idéntica a (2.2), con lo cual la omitiremos para no duplicarlas.

Por otro lado, si se utiliza la actividad  $a = (f, e_{end}^c)$ , siendo  $f$  el último evento de alguna tarea  $\theta = [[e, f]]$ , la restricción (3.5) correspondiente garantiza que  $x_f \leq w_c^{off}$ , es decir que la última tarea que realiza  $c$  termine antes de  $w_c^{off}$ . Cuando  $z_a = 0$

tomamos  $\bar{M}_f^c \stackrel{\text{def}}{=} q_f + d_f - w_c^{off}$  para que la desigualdad se cumpla sin importar el valor de  $x_f$ . Notar que si  $q_f + d_f = w_c^{off}$ , la restricción es implicada por (2.3), con lo cual la omitiremos ya que es redundante.

Las restricciones (3.6), (3.7) y (3.8) definen las reglas del flujo de tripulación en la red. En particular las restricciones (3.6) están asegurando que la capacidad de cada tripulación es 1, de manera similar a las (2.7) de los demás recursos.

Las restricciones (3.7) son las que preservan el flujo de una tripulación al transferirse de una tarea a la otra. Del lado izquierdo se cuenta la cantidad de veces que la tripulación  $c$  se asigna a la tarea  $\theta$  y del lado derecho la cantidad de veces que dicha tarea traspasa la tripulación  $c$ . Por las restricciones anteriores sabemos que ambos lados de la igualdad pueden valer 0 o 1, es decir que las únicas opciones válidas son que o bien  $c$  sea asignado a  $\theta$  y luego traspasado, o bien que  $c$  no sea ni asignado ni traspasado.

Por último las restricciones (3.8) aseguran que si  $c$  es utilizada en alguna tarea, debe regresar a la base. Del lado izquierdo contamos la cantidad de veces que el evento de inventario inicial asignó  $c$  a alguna tarea (0 o 1) y del lado derecho la cantidad de veces que el evento final recibe a  $c$  (también 0 o 1, por la conservación del flujo de las restricciones anteriores). O bien ambos son cero y  $c$  no fue utilizado, o bien ambos son 1 y  $c$  entonces  $c$  regresó a su base.

Las restricciones (3.9) aseguran que todas las tareas de los servicios no cancelados tengan tripulación para circular. Del lado derecho se cuenta la cantidad de tripulaciones asignadas a la tarea  $\theta$  (por las restricciones anteriores, cada tripulación no puede asignarse más de una vez). Si el tren asociado a  $\theta$  no se cancela, la desigualdad pide que haya al menos una tripulación asignada a  $\theta$ . Si en cambio el tren se cancela, la restricción se cumple trivialmente (siempre es no negativo el lado derecho).

Por otro lado las restricciones (3.10) limitan la cantidad de tripulaciones que pueden ser asignadas a una tarea. Recordemos que en las tareas de **reposicionamiento**, las tripulaciones podían viajar como pasajeras de un servicio de la misma línea. Entonces es necesario permitir más de una tripulación en la misma tarea. De esta manera, asumimos que alguna de las tripulaciones realiza la tarea de manejo y el resto la utilizan como reposicionamiento. Esta forma de modelar tareas de reposicionamiento sin distinguirlas explícitamente en el modelo lineal fue utilizada también por Potthoff y Huisman [4], Potthoff [5] y Huisman [7]. El parámetro  $P_{\text{pas}}$  indica la cantidad máxima de tripulaciones como pasajeras de una tarea. Luego, si el tren de la tarea  $\theta$  no se cancela, la restricción asegura que no se asignan a  $\theta$  más de  $P_{\text{pas}}$  tripulaciones pasajeras y una conduciendo. Si en cambio el tren se cancela, esta restricción fuerza a que ninguna tripulación sea asignada.

### 3.4.2. Función Objetivo

Como mencionamos en la Sección 3.1, distintos aspectos son tenidos en cuenta a la hora de elegir qué optimizar. El primero de ellos es la minimización del impacto al usuario, es decir las cancelaciones y demoras. Este aspecto ya fue cubierto con la función objetivo del modelo de reprogramación del cronograma ferroviario y sigue siendo válido luego de integrar las restricciones de tripulación.

A continuación analizaremos los otros dos aspectos mencionados en dicho capítulo, que son los cambios a la jornada de trabajo original y los costos operacionales.

### Cambios con respecto a la jornada original

Demasiados cambios en la jornada laboral de cada tripulación puede llevar a dos problemas, ambos debido a que la comunicación de los mismos suele realizarse de forma manual. Por un lado, en Dollevoet et al. [3] se menciona que puede suceder que se necesite demasiado tiempo para informar todas estas modificaciones, haciendo infactible esta opción ya que la red sigue cambiando y la reacción debe ser lo más rápida posible. Y por otro lado, en Potthoff y Huisman [4] y Potthoff [5] se menciona que los errores de comunicación son muy comunes y pueden inducir un peor estado en la red.

Para minimizar los cambios realizados en las jornadas alternativas con respecto a la jornada original, primero debemos definir matemáticamente un *cambio* y luego penalizarlo en la función objetivo. Sea entonces  $\mathbb{O}_{\theta,c} \in \{0,1\}$  una constante que vale 1 si  $c$  realizaba la tarea  $\theta$  en su jornada original, ya sea conduciendo o como reposicionamiento. Definiremos las siguientes variables:

$$\omega_{\theta,c} = \begin{cases} 1 & \text{si } c \text{ realiza } \theta \text{ en la jornada alternativa} \\ & \text{ya sea conduciendo o como reposicionamiento.} \\ 0 & \text{cc} \end{cases}$$

$$\delta_{\theta,c} = \begin{cases} 1 & \text{si } \mathbb{O}_{\theta,c} = 1 \wedge \omega_{\theta,c} = 0 \wedge y_{t_\theta} = 0 \\ 0 & \text{cc} \end{cases}$$

Notar que  $\delta_{\theta,c}$  es la variable que define un *cambio* en la jornada laboral y será utilizada para penalizarlos.  $\delta_{\theta,c} = 1$  cuando  $\theta$  era parte de la jornada original de  $c$ , pero no es parte de la jornada alternativa, y el tren correspondiente no fue cancelado. Si el tren fuera cancelado no queremos penalizar el cambio porque ya estamos penalizando la cancelación en sí. También podríamos penalizar el caso opuesto, i.e. que  $c$  no realizaba la tarea originalmente pero sí la realiza en la jornada nueva, pero no lo podemos hacer porque no queremos penalizar que una tripulación realice nuevas tareas o tareas de pasajero (que ya se penalizan por separado), y además podríamos estar penalizando dos veces el mismo intercambio de tareas. El esquema elegido sirve para penalizar a cada tripulación por cada tarea que deja de realizar.

Para definir las variables, primero notar que:

$$\omega_{\theta,c} = \sum_{a \in A_{cr}^-(c,\theta)} z_a$$

Por restricciones (3.6) y (3.7) sabemos que la sumatoria no puede valer más que 1 (no se puede asignar más de una vez  $c$  a  $\theta$ ). El resultado de la suma indica si efectivamente se realizó la asignación o no. Luego podemos definir la variable de cambio como:

**Prop. 3.2.** *La variable de cambio se puede definir como*

$$\delta_{\theta,c} = \mathbb{O}_{\theta,c}(1 - y_{t_\theta} - \omega_{\theta,c})$$

*Demostración.* Por un lado observar que la igualdad es lineal en las variables. Veamos que además cumple con la definición deseada de  $\delta_{\theta,c}$ .

- $\mathbb{O}_{\theta,c} = 0 \Rightarrow \delta_{\theta,c} = 0$
- $\mathbb{O}_{\theta,c} = 1$ 
  - $y_{t_\theta} = 1 \Rightarrow \omega_{\theta,c} = 0$  (por restricciones (3.10) ninguna tripulación será asignada a  $\theta$ ). Luego  $\delta_{\theta,c} = 0$
  - $y_{t_\theta} = 0$

$$\begin{aligned} \circ \omega_{\theta,c} = 1 &\Rightarrow \delta_{\theta,c} = 0 \\ \circ \omega_{\theta,c} = 0 &\Rightarrow \delta_{\theta,c} = 1 \end{aligned}$$

□

Luego agregamos al modelo PLE las restricciones para definir  $\delta_{\theta,c}$ :

$$\delta_{\theta,c} = \mathbb{O}_{\theta,c} \left( 1 - y_{t_\theta} - \sum_{a \in A_{cr}^-(c,\theta)} z_a \right) \quad \forall c \in \mathbb{C}, \theta \in \Theta \quad (3.11)$$

$$\delta_{\theta,c} \in \{0, 1\} \quad \forall c \in \mathbb{C}, \theta \in \Theta \quad (3.12)$$

Observar que reemplazamos  $\omega_{\theta,c}$  por su definición, ya que no es necesario agregarla al modelo.

Finalmente, agregamos a la función objetivo (2.1) el término:

$$W_\delta \sum_{\theta \in \Theta} \sum_{c \in \mathbb{C}} \delta_{\theta,c}$$

donde  $W_\delta$  es un peso que asignaremos a esta penalización como parámetro del problema.

Una mejora que hicimos al momento de implementar esta parte del modelo es agregar sólo las restricciones y las variables cuando  $\mathbb{O}_{\theta,c} = 1$ , ya que caso contrario  $\delta_{\theta,c} = 0$  y no hay ninguna penalización, independientemente del valor de las variables.

### Costos operacionales

Según Potthoff y Huisman [4] los reposicionamientos de tripulación dentro de la red (viajando como pasajera de otra tarea) no generan un costo adicional para la empresa operadora (como sí ocurre en la aviación por ejemplo). Sin embargo, en nuestro modelo vamos a incluir una penalización a estos reposicionamientos (con un costo bajo) para minimizar el uso innecesario de la tripulación y de los reposicionamientos, y darle más flexibilidad al modelo y la experimentación.

Además más adelante vamos a flexibilizar algunas restricciones, por ejemplo permitiendo a una tripulación terminar fuera de su horario de fin o en una estación que no es su base, incurriendo en costos de horas extras y viajes en taxi para regresar a la base. Referirse a las secciones 3.7.1 y 3.7.2 para el análisis de estos casos respectivamente.

Entonces para minimizar los reposicionamientos dentro de la red, debemos minimizar la cantidad de tripulaciones que viajan como pasajeras en cada tarea. Para esto debemos definir una variable  $p_\theta \in \{0, \dots, \mathbb{P}_{pas}\}$  para cada  $\theta \in \Theta$  como la cantidad de tripulaciones que viajan como pasajeras en la tarea  $\theta$  agregando el siguiente conjunto de restricciones:

$$\left( \sum_{a \in A_{cr}^-(\theta)} z_a \right) - 1 \leq p_\theta \quad \forall \theta \in \Theta \quad (3.13)$$

Y además agregamos a la función objetivo (2.1) el término

$$W_p \sum_{\theta \in \Theta} p_\theta$$

que será minimizado al resolver el problema, siendo  $W_p$  el peso.

Observar que la sumatoria de (3.13) es no negativa y cuenta la cantidad de tripulaciones asignadas a la tarea  $\theta$ , con lo cual si la restricción se cumpliera por igualdad,  $p_\theta$  estaría tomando el valor deseado. Dado que estamos minimizando el valor de  $p_\theta$ , siempre se cumplirá por igualdad salvo en el caso en que el tren  $t_\theta$  sea cancelado, en el cual  $p_\theta = 0$  (el mínimo valor posible) y la desigualdad será estricta (ya que la sumatoria toma valor nulo).

Notar también que no estamos teniendo en cuenta la duración de las tareas para penalizarlas, i.e. tiene el mismo costo que una tripulación sea pasajera de una tarea de 1 minuto o de 90 minutos. Esto se podría modificar tomando un costo distinto por cada tarea, proporcional a la duración, pero no agregaría complejidad al problema.

### 3.4.3. Horarios de almuerzo

Como vimos en la Sección 3.1, una de las condiciones más restrictivas para que una jornada alternativa sea válida es que debe contemplar un horario de almuerzo de una duración mínima y debe garantizar que no se realice ni demasiado temprano ni demasiado tarde en la jornada. No tenerla en cuenta podría significar más cancelaciones en la etapa de reasignación de tripulación del framework iterativo que vimos en el Capítulo 1.

Para modelar estas restricciones, primero debemos agregar algunas aristas a la red  $N_{\mathbb{C}}(\Theta)$ . El almuerzo sólo puede ser realizado en estaciones de relevo  $s \in S^{rel}$  con una duración mínima de  $\mathbb{P}_{mb}$ , que asumiremos mayor o igual al tiempo mínimo de conexión entre tareas  $\mathbb{P}_{conn}$ . Esto significa que podemos modelar el almuerzo como una conexión con una duración mínima mayor o igual, es decir que las actividades de almuerzo serán un subconjunto de las de conexión.

Sea  $A_{mb}$  un conjunto de actividades que permiten tomar almuerzo. La mínima duración de la actividad será el tiempo mínimo necesario para realizarlo, i.e.  $L_a = \mathbb{P}_{mb}$ . Sea  $a = (e, f) \in A_{cr}$ , agregaremos  $\bar{a} = (e, f)$  con  $L_{\bar{a}} = \mathbb{P}_{mb}$  si:

$$q_e + \mathbb{P}_{mb} \leq q_f + d_f$$

Observar que la condición es análoga a la segunda condición de una arista en el grafo de alcance (3.3) cambiando el tiempo de conexión por el tiempo de almuerzo. La idea de fondo en esta desigualdad, como explicamos anteriormente, es que  $x_e + \mathbb{P}_{mb} \leq x_f$ , i.e. que haya tiempo suficiente entre los eventos  $e$  y  $f$  para realizar el recreo de almuerzo. Pero por ser  $x_e, x_f$  variables, debemos acotar sus valores y agregar  $\bar{a}$  si existe la posibilidad de que la condición sea válida con los horarios reprogramados.

Las actividades de almuerzo de cada tripulación  $A_{mb}(c)$  se agregan al conjunto original de actividades de tripulación  $A_{cr}(c)$ , y por ende se agregan todas también a  $A_{cr}$ . El resultado es que algunas actividades quedan *duplicadas* con distintos valores de  $L_a$  (una el tiempo de conexión y la otra el tiempo de almuerzo). En la Figura 3.11 vemos un ejemplo con las posibles actividades de tripulación entre dos pares de tareas  $(\theta_1, \theta_2)$  y  $(\theta_1, \theta_3)$ . Suponiendo un tiempo mínimo de conexión de 5 minutos y una duración mínima de almuerzo de 30 minutos, y suponiendo que no podemos demorar los horarios de las tareas, en el primer par vemos que no es posible realizar un almuerzo entre  $\theta_1$  y  $\theta_2$ , pero sí es posible una conexión. En cambio entre  $\theta_1$  y  $\theta_3$ , ambos tipos de traspaso son posibles.

Esta modificación a  $N_{\mathbb{C}}(\Theta)$  impacta en las restricciones que ya definimos. En las restricciones de conservación de flujo (3.7) las sumatorias realizadas sobre conjuntos de actividades ahora estarán incluyendo también a las de almuerzo, lo cual va a garantizar que una tripulación  $c$  que puede traspasarse a  $\theta$  con o sin almuerzo, sólo pueda elegir una de estas opciones (ya que la capacidad de  $c$  es 1 y ambos arcos corresponden a  $c$ ). Es decir que se sigue garantizando el correcto funcionamiento del flujo. Y de esta manera, en las restricciones de mínima (3.9) y máxima (3.10)

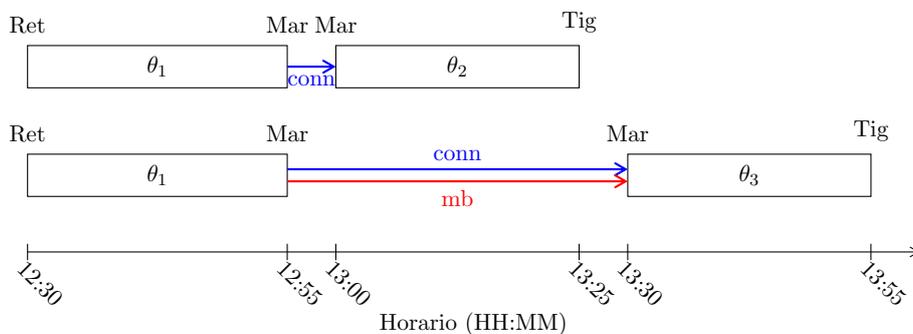


Fig. 3.11: Ejemplo de las posibles actividades de tripulación entre dos tareas: conexión (*connection*) y almuerzo (*meal break*). Asume  $\mathbb{P}_{\text{conn}} = 5$  minutos y  $\mathbb{P}_{\text{mb}} = 30$  minutos y que no es posible demorar partidas o arribos.

cantidad de tripulación por tarea se asegura que no se va a contar dos veces a la misma tripulación.

Por otro lado, en el modelo original de reprogramación, en las restricciones (2.8) se garantiza que al usar una actividad de almuerzo, debe transcurrir el tiempo mínimo  $L_a = \mathbb{P}_{\text{mb}}$  desde finalizar la tarea anterior hasta empezar la tarea posterior al almuerzo. También se agregan las variables  $z_a$  para  $a \in A_{mb}$  en (2.9).

Pero además vamos a agregar las restricciones del horario de almuerzo al modelo. Para la condición de que no se realice demasiado temprano ni demasiado tarde tenemos dos parámetros:  $\tau_1^{mb}$  indica el tiempo máximo entre que  $c$  inicia su jornada ( $w_c^{on}$ ) y el inicio del almuerzo, y  $\tau_2^{mb}$  el tiempo máximo entre que finaliza el almuerzo y  $w_c^{off}$ . Es decir que para cada  $c \in \mathbb{C}$ ,  $\tau_1^{mb}$  define el intervalo de tiempo en el cual puede comenzar el almuerzo (relativo a  $w_c^{on}$ ) y  $\tau_2^{mb}$  el intervalo en el cual puede terminar el almuerzo (relativo a  $w_c^{off}$ ). Si  $x_e$  y  $x_f$  son el comienzo y el fin del horario de almuerzo respectivamente, vamos a agregar restricciones tal que:

$$x_e \in [w_c^{on}, w_c^{on} + \tau_1^{mb}] \quad \text{y} \quad x_f \in [w_c^{off} - \tau_2^{mb}, w_c^{off}]$$

La idea de estas restricciones es que el almuerzo no *comience* demasiado tarde y no *finalice* demasiado temprano. Por ejemplo, en una jornada de 10:00 a 18:00 horas, podríamos pedir que el almuerzo comience antes de las 15:00 y que termine después de las 13:00 ( $\tau_1^{mb} = \tau_2^{mb} = 5hs$ ). Entonces un horario de almuerzo válido sería por ejemplo de 12:30 a 13:00, o de 15:00 a 15:30, u otros rangos horarios intermedios, pero no sería válido un almuerzo de 12:00 a 12:50 (termina demasiado temprano) o de 15:10 a 16:00 (empieza demasiado tarde). Observar que no importa la longitud del horario asignado,  $L_a$  especifica una duración mínima del almuerzo, pero no máxima. Siguiendo el ejemplo, sería válido un horario de almuerzo de 11:00 a 16:00, ya que da libertad a la tripulación de tomar el almuerzo dentro de ese rango que no empieza muy tarde y no termina muy temprano. Incluso podríamos asignar un horario de almuerzo de 10:00 a 18:00, i.e. que la tripulación no realice ninguna tarea y sólo tome el almuerzo. Es decir que sin importar el valor de los parámetros  $\tau_1^{mb}$  y  $\tau_2^{mb}$ , siempre existe un horario de almuerzo válido. La infactibilidad del almuerzo de  $c$  estará asociada a las posibles actividades de almuerzo en la red  $N_c(\Theta)$ .

A continuación presentamos las restricciones de almuerzo:

$$\sum_{a \in A_{mb}(c)} z_a = \sum_{a \in A_{cr}^+(e_{beg}^c)} z_a \quad \forall c \in \mathbb{C} \quad (3.14)$$

$$x_e - w_c^{on} - M_e^c(1 - z_a) \leq \tau_1^{mb} \quad \forall c \in \mathbb{C}, a = (e, f) \in A_{mb}(c) \quad (3.15)$$

$$w_c^{off} - x_f - \bar{M}_f^c(1 - z_a) \leq \tau_2^{mb} \quad \forall c \in \mathbb{C}, a = (e, f) \in A_{mb}(c) \quad (3.16)$$

$A_{mb}(c)$	conjunto de actividades de almuerzo de $c$
$\tau_1^{mb}$	Máximo tiempo entre $w_c^{on}$ y el comienzo del almuerzo
$\tau_2^{mb}$	Máximo tiempo entre el fin del almuerzo y $w_c^{off}$

Tab. 3.2: Notación restricciones de almuerzo.

Las igualdades (3.14) fuerzan a que  $c$  deba realizar exactamente una actividad de almuerzo si es utilizado en alguna tarea. Si  $c$  no fuera utilizado en ninguna tarea, quiere decir que  $c$  tiene una jornada de reserva y en tal caso nuestro modelo no lo tendrá en cuenta para asignarle horario de almuerzo (asumimos que tendrá tiempo suficiente para hacerlo). El lado izquierdo de la desigualdad cuenta las actividades de almuerzo de  $c$  utilizadas en la solución y el lado derecho la cantidad de actividades iniciales de  $c$ , que pueden ser 0 o 1 según las restricciones de capacidad (3.6).

Las desigualdades (3.15) y (3.16) aseguran que el almuerzo no comience demasiado tarde ni finalice demasiado temprano, respectivamente. Supongamos que  $z_a = 1$  con  $a = (e, f) \in A_{mb}(c)$ , luego estas desigualdades deben verificarse:

$$x_e - w_c^{on} \leq \tau_1^{mb} \quad \text{y} \quad w_c^{off} - x_f \leq \tau_2^{mb}$$

Por las restricciones (3.4) y (3.5), y por la construcción del grafo de alcance  $G^R(\Theta)$ , sabemos que todas las tareas o eventos que realiza una tripulación estarán dentro de su ventana de disponibilidad, con lo cual  $w_c^{on} \leq x_e$  y  $x_f \leq w_c^{off}$ . Con las dos restricciones de almuerzo, quedaría que:

$$x_e \in [w_c^{on}, w_c^{on} + \tau_1^{mb}] \quad \text{y} \quad x_f \in [w_c^{off} - \tau_2^{mb}, w_c^{off}]$$

siendo  $x_e$  el horario de inicio del almuerzo y  $x_f$  el horario de fin.

En el caso de  $z_a = 0$ , i.e. para las actividades de almuerzo que no fueron utilizadas, las desigualdades (3.15) y (3.16) se cumplen independientemente del valor de las variables definiendo:

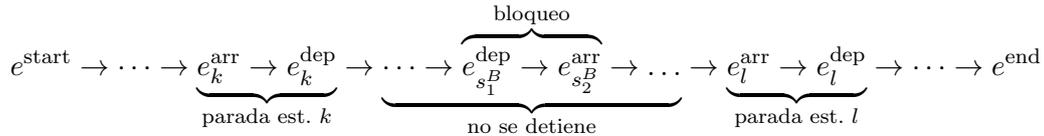
$$M_e^c \stackrel{\text{def}}{=} q_e + d_e - w_c^{on} - \tau_1^{mb}$$

$$\bar{M}_f^c \stackrel{\text{def}}{=} w_c^{off} - q_f - \tau_2^{mb}$$

### 3.5. Bloqueo de vías

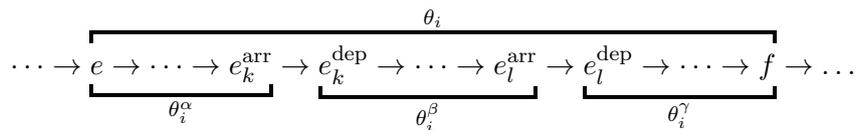
Al igual que en el Capítulo 2, para modelar correctamente el bloqueo de vías debemos hacer algunos cambios en la red  $N_C(\Theta)$ , i.e. en el conjunto de actividades de tripulación. Esto nos llevará a tener que agregar algunas restricciones para garantizar la correcta asignación de recursos en la zona del bloqueo.

En la Sección 2.5.1 explicamos que el bloqueo se produce en un tramo de vías  $(s_1^B, s_2^B)$ , con  $s_1^B, s_2^B \in S$ , y cada servicio que pasaba por la zona del bloqueo  $t \in T^{\text{disr}} \subset T$  debía partirse en  $\alpha_t, \beta_t, \gamma_t$ , tomando como estaciones de partición la última estación  $k$  donde el tren se detiene antes del bloqueo y la primer estación  $l$  donde el tren se detiene después del bloqueo. Podría suceder que  $k = s_1^B$  y/o  $l = s_2^B$  si el servicio paraba en alguna de esas estaciones en el cronograma original.



Sea  $t = [\theta_1, \dots, \theta_n]$  la partición en tareas de los eventos del servicio  $t \in T^{\text{disr}}$ . Como mencionamos en la Sección 3.2, las tareas están definidas entre estaciones de relevo  $S^{\text{rel}}$  donde el tren se detiene. Dado que el bloqueo se produce en una sección de vías  $(s_1^B, s_2^B)$  y por ende no hay estaciones intermedias, sabemos que existe alguna tarea  $\theta_i = [[e, f]]$  que contiene a ese tramo. Entonces realizaremos una partición de  $\theta_i$  en  $\theta_i^\alpha, \theta_i^\beta, \theta_i^\gamma$  de la siguiente manera:

- $\theta_i^\alpha = [[e, e_k^{\text{arr}}]]$ , siendo  $e_k^{\text{arr}}$  el arribo a la estación  $k$ , i.e. donde finaliza  $\alpha_t$ .
- $\theta_i^\beta = [[e_k^{\text{dep}}, e_l^{\text{arr}}]]$ , siendo  $e_k^{\text{dep}}$  la salida de la estación  $k$  y  $e_l^{\text{arr}}$  el arribo a la estación  $l$ , i.e.  $\beta_t = [\theta_i^\beta]$ .
- $\theta_i^\gamma = [[e_l^{\text{dep}}, f]]$ , siendo  $e_l^{\text{dep}}$  la salida desde la estación  $l$ , i.e. donde comienza  $\gamma_t$ .



Observar que  $\theta_i^\beta$  siempre existe y es la única tarea del servicio  $\beta_t$ . Sin embargo,  $\theta_i^\alpha$  podría no tener ningún evento si por ejemplo  $e = e_k^{\text{dep}}$ , es decir si  $\theta_i$  empieza donde empieza  $\beta_t$ . Análogamente  $\theta_i^\gamma$  podría no tener ningún evento si  $f = e_l^{\text{arr}}$ , es decir si la tarea  $\theta_i$  termina donde termina  $\beta_t$ .

En conclusión, al realizar la partición de  $t$ , quedan los siguientes trenes y tareas:

$$\alpha_t = [\theta_1, \dots, \theta_i^\alpha], \quad \beta_t = [\theta_i^\beta], \quad \gamma_t = [\theta_i^\gamma, \dots, \theta_n]$$

El problema que aparece al realizar la partición es que las nuevas tareas podrían tener uno o ambos extremos en estaciones que no son de relevo. Recordemos que no todas las estaciones donde el tren se detiene son necesariamente de relevo, y dado que la partición se hace en estaciones de parada, no hay garantía de que sean de relevo también. Esto significa que en dichas estaciones no se podrán realizar trasposos de tripulación que impliquen un cambio de material rodante, i.e. la tripulación no puede salir del material rodante. Esto motiva la restricción (3.17) que definiremos a continuación.

Notar que cuando introdujimos el grafo de alcance en la Sección 3.2.2, intencionalmente definimos (3.2) sobre el conjunto de estaciones  $S$  y no sobre el subconjunto  $S^{rel}$ . De esta manera, al agregar estas tareas especiales al modelo, el grafo de alcance sigue estando bien definido. Además recordar que las aristas del grafo de alcance se trasladan a  $N_C(\Theta)$ , i.e. a las actividades de tripulación  $A_{cr}$  y a la red de evento-actividad extendida.

Supongamos que  $a_c \in A_{cr}$  traspasa un recurso de tripulación de  $e$  (evento final de una tarea  $\theta_i$ ) a  $f$  (evento inicial de una tarea  $\theta_j$ ), siendo  $k \notin S^{rel}$  la estación en la que ocurre el traspaso. Es decir que  $a_c$  es una actividad de tripulación que conecta dos tareas de la partición en la zona de bloqueo ( $\theta^\alpha, \theta^\beta, \theta^\gamma$ ) entre sí. Como  $k$  no es de relevo, la tripulación no puede realizar una conexión en  $k$ , i.e. si en la reasignación elegimos usar  $a_c$  para traspasar la tripulación de  $e$  a  $f$ , luego debemos también traspasar el recurso de material rodante de  $e$  a  $f$ . Supongamos que dicho traspaso existe y llamémosle  $a_r \in A_{rol}$ . Notar que  $a_c$  y  $a_r$  son actividades de traspaso de distintos recursos entre el mismo par de eventos. Luego debemos agregar una restricción a nuestro modelo que fuerce a que  $a_r$  sea utilizada en la solución si  $a_c$  es utilizada ( $z_{a_c} \leq z_{a_r}$ ), es decir que se traspase material rodante si se necesita traspasar la tripulación en una estación que no es de relevo.

Entonces definimos el conjunto de pares  $P_{rol}^{cr} \subset A_{cr} \times A_{rol}$  tal que  $(a_c, a_r) \in P_{rol}^{cr}$  si  $a_c$  requiere un traspaso de material rodante de acuerdo a la descripción anterior y  $a_r$  es el traspaso de material rodante asociado. Formalmente:

$$(a_c, a_r) \in P_{rol}^{cr} \Leftrightarrow e \in E_{arr}^k, f \in E_{dep}^k, k \notin S^{rel}$$

$$\text{tq } a_c = (e, f) \in A_{cr} \text{ y } a_r = (e, f) \in A_{rol}$$

Observar que una misma actividad de material rodante podría aparecer en más de un par, ya que entre  $e$  y  $f$  hay una actividad de tripulación por cada  $c$  que pueda realizarla (según la construcción de  $N_C(\Theta)$ ).

Pero esta definición está dejando fuera los casos en los que no existe una actividad de material rodante asociada. Esto pasa por ejemplo porque los trenes  $\beta_t$  sólo pueden tomar y transferir material rodante desde/hacia sus respectivos trenes  $\alpha_t$  y  $\gamma_t$ , mientras que el recurso de tripulación pueden traspasarlo a cualquier otra tarea  $\theta^\alpha, \theta^\beta, \theta^\gamma$  (que cumpla con las condiciones de alcance de  $G^R(\Theta)$ ). Para estos casos definiremos el conjunto  $P_{rol}^{cr} \subset A_{cr}$  tal que  $a_c \in P_{rol}^{cr}$  si  $a_c$  requiere un traspaso de material rodante pero no existe tal traspaso. Notar que no es un conjunto de pares ya que justamente no tenemos la segunda componente.

Luego agregamos las siguientes restricciones al modelo PLE:

$$z_{a_c} \leq z_{a_r} \quad \forall (a_c, a_r) \in P_{rol}^{cr} \quad (3.17)$$

$$z_{a_c} = 0 \quad \forall a_c \in P_{rol}^{cr} \quad (3.18)$$

Las restricciones (3.17) aseguran si la actividad de tripulación  $a_c$  es utilizada, la actividad de material rodante asociada  $a_r$  también debe ser utilizada. En cambio si  $a_c$  no es utilizada,  $a_r$  puede o no ser utilizada (ya que el traspaso de material rodante podría realizarse de todas formas y la tripulación traspasarla con otra actividad).

Por otro lado, las restricciones (3.18) fuerzan a que el modelo no utilice las actividades  $a_c$  que requieren traspaso de material rodante, pero dicho traspaso no existe. En esencia es la misma restricción que la anterior cuando  $a_r$  no se utiliza (i.e.  $z_{a_r} = 0$ ), ya que en este caso ni siquiera existe tal  $a_r$ .

### 3.6. Recorte temporal a la red de flujo de tripulación

Como vimos en la Sección 2.5.2, no todos los eventos se incluyen en el modelo, ya que por un lado todos los eventos que ocurrieron antes de  $\tau_1$  ya no pueden cambiarse y todos los que están programados después de  $\tau_3$  no permitiremos que se demoren. Omitimos los detalles de los distintos casos borde, pero pueden ser consultados con un gran nivel de detalle en Mosteiro [2].

En el caso de la red de flujo de tripulación  $N_{\mathbb{C}}(\Theta)$  también vamos a eliminar algunas tareas que no son necesarias para reasignar tripulación. En particular veremos que antes de  $\tau_1$  podemos eliminar tareas con un criterio similar al utilizado para eliminar eventos, pero para las tareas posteriores a  $\tau_3$  realizaremos algunas transformaciones a la red para eliminar una cantidad considerable de nodos sin perder opciones que nos dan mayor factibilidad.

Por último cabe destacar que en los casos en los que tengamos tareas de tripulación cuyos eventos fueron total o parcialmente excluidos del modelo según Mosteiro [2], fijaremos todas las variables correspondientes y las trataremos como constantes a la hora de armar las restricciones. En particular un evento  $e$  que es excluido, verifica que  $x_e = q_e$  y  $d_e = 0$ . De esta manera evitamos tener variables innecesarias y resulta más simple la integración con el resto del modelo.

#### 3.6.1. Tareas previas al inicio del bloqueo

Los eventos  $e$  que ocurrieron antes de  $\tau_1$ , i.e.  $q_e < \tau_1$  cumplen que  $x_e = q_e$ ,  $d_e = 0$  y  $y_{te} = 0$ , es decir que no se demoran ni se cancelan, porque ya ocurrieron.

La red de flujo de tripulación  $N_{\mathbb{C}}(\Theta)$  fue creada para modelar el uso del recurso de tripulación, o dicho en otras palabras, la asignación secuencial de cada tripulación a las tareas. Antes del inicio del bloqueo hay tareas que ya se completaron y liberaron a la/s tripulación/es que tenían asignada/s en el esquema original. En una primera aproximación desearíamos eliminar todas las tareas que finalizaron antes de  $\tau_1$ . El problema con esto es que si una tripulación terminó por ejemplo 1 minuto antes de  $\tau_1$  y el mínimo tiempo de conexión es de 10 minutos, la tripulación estará disponible para realizar su próxima tarea en  $\tau_1 + 9$ . Si simplemente eliminamos la tarea anterior, tendríamos que modificar el modelo actual para incluir este caso. En vez de esto, vamos a establecer un límite anterior a  $\tau_1$ , antes del cual será *seguro* eliminar tareas.

Cuando una tripulación pasa de una tarea a la otra ocurre una conexión o un recreo de almuerzo. En cualquiera de estos casos, si antes de  $\tau_1$  ya transcurrió el tiempo mínimo de conexión o almuerzo (resp.), es seguro eliminar la tarea anterior para esta tripulación. En cambio si en  $\tau_1$  aún no terminó ese tiempo mínimo de conexión o almuerzo, entonces no es seguro eliminarla, ya que el modelo debe contener ese traspaso y hacer respetar el tiempo mínimo. Entonces sea

$$\text{lím}_c^- = \tau_1 - \text{máx}\{\mathbb{P}_{\text{conn}}, \mathbb{P}_{\text{mb}}\}$$

el tiempo de seguridad para eliminar tareas antes de  $\tau_1$ . Observar que  $\text{lím}_c^- = \tau_1 - \mathbb{P}_{\text{mb}}$  ya que en la Sección 3.4.3 asumimos que  $\mathbb{P}_{\text{mb}} \geq \mathbb{P}_{\text{conn}}$ . Todas las tareas  $\theta = [[e, f]]$  tales que  $q_f \leq \text{lím}_c^-$  son eliminadas del modelo. Supongamos que la tarea termina justo en el límite, es decir que  $q_f = \text{lím}_c^-$ . Si existe una tripulación  $c$  asignada originalmente a esa tarea que tomaba su horario de almuerzo luego de finalizarla,  $c$  estaría disponible justo en  $\tau_1$ . Si la tarea terminaba estrictamente antes,  $c$  estará disponible antes de  $\tau_1$ , pero como no comenzó otra tarea después de eso, está disponible en  $\tau_1$  también. En la Figura 3.12 se presentan distintos ejemplos de tareas anteriores a  $\tau_1$  en las jornadas originales de las tripulaciones. Tanto  $c_1$  como  $c_2$  se encontrarán disponibles en  $\tau_1$  en las estaciones donde terminaron su última tarea, ya que el tiempo entre  $\text{lím}_c^-$  y  $\tau_1$  asegura que ya completaron el tiempo mínimo para conexión o almuerzo, con lo cual podemos eliminar de forma segura  $\theta_1$  y  $\theta_2$ .

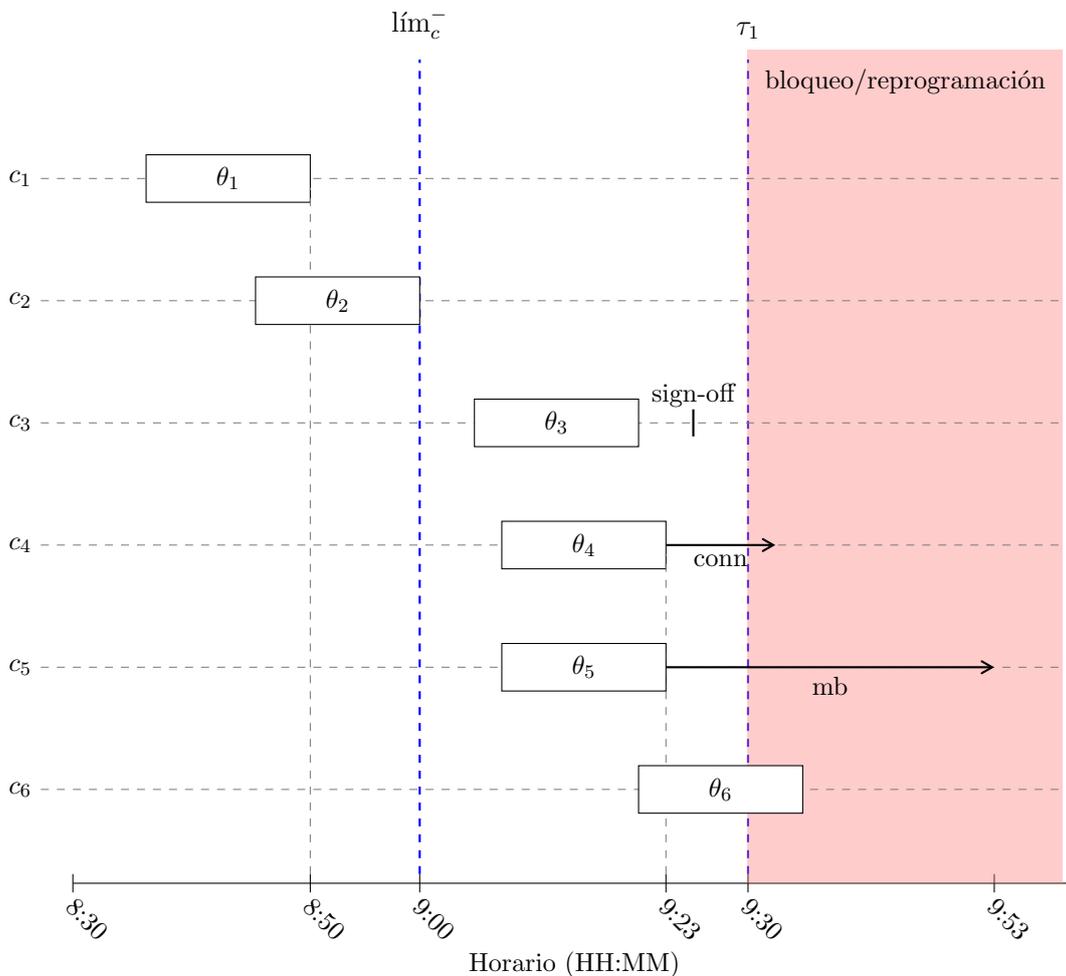


Fig. 3.12: Ejemplos de tareas previas a  $\tau_1$  en jornadas originales de distintas tripulaciones. Tomamos  $\mathbb{P}_{\text{conn}} = 10$  minutos y  $\mathbb{P}_{\text{mb}} = 30$  minutos, lo cual define  $\text{lim}_c^- = \tau_1 - 30$ . Asumimos que cada tarea sólo tiene asignada la tripulación correspondiente en la jornada original y ninguna otra.

Además de eliminar dichas tareas, vamos a eliminar del modelo todas las tripulaciones que en  $\tau_1$  ya terminaron su jornada, es decir tales que  $w_c^{\text{off}} < \tau_1$ , ya que son recursos que no están disponibles durante la reprogramación. Luego, si existe una tarea  $\theta = [[e, f]]$  cuyo horario de finalización  $q_f \in (\text{lim}_c^-, \tau_1]$ , i.e. ya finalizó pero fue incluida en el modelo, pero resulta que todas las tripulaciones que realizaban  $\theta$  en su jornada original fueron eliminadas del modelo, en tal caso  $\theta$  también será eliminada, ya que no traspasa ninguna tripulación. Por ejemplo este es el caso de  $c_3$  en la Figura 3.12, y por lo tanto podemos eliminar  $\theta_3$  ya que  $c_3$  es la única tripulación asignada a esa tarea en el esquema original.

Por último observar que  $\theta_4$ ,  $\theta_5$  y  $\theta_6$  de la Figura 3.12 no pueden ser eliminadas de acuerdo a  $\text{lim}_c^-$ . A continuación analizaremos en detalle estos casos.

### Adaptaciones a la red de flujos de tripulación

Al eliminar todas las tareas que mencionamos anteriormente, es necesario realizar algunos ajustes a la definición de la red de tripulación  $N_{\mathbb{C}}(\Theta)$ . Si pensamos  $N_{\mathbb{C}}(\Theta)$  como la unión de las redes individuales de cada tripulación  $N_c(\Theta)$  para cada  $c \in \mathbb{C}$ , las tareas eliminadas siempre son previas a  $\tau_1$ , con lo cual podría suceder que algunas o todas las tareas iniciales de  $c$  se hayan eliminado, o que algunas hayan perdido sentido para  $c$  (en el caso de que empezaron antes de  $\tau_1$  pero  $c$  no fue asignada a la misma). Si  $w_c^{\text{on}} \geq \tau_1$ ,  $c$  no empezó o está empezando su jornada laboral, con lo cual

no se ve afectada por la eliminación de tareas. Veamos entonces cómo se modifica  $N_c(\Theta)$  para cada  $c$  tal que  $w_c^{on} < \tau_1$ .

Si  $c$  no había empezado a realizar ninguna tarea antes de  $\tau_1$ , está disponible en su estación base en  $\tau_1$  y sus posibles tareas iniciales serán las que estaban definidas en  $N_c(\Theta)$  salvo las que eliminamos previamente del modelo. En el caso de que no haya quedado ninguna tarea inicial posible para  $c$ , simplemente no será utilizado en la solución.

Por el contrario, si  $c$  había empezado a realizar alguna tarea antes de  $\tau_1$ , debemos redefinir el conjunto de posibles tareas iniciales de  $c$ , i.e.  $A_{cr}^+(e_{beg}^c)$ . Observar que  $c$  pudo haber empezado y completado una o más tareas antes de  $\tau_1$ , pero en tal caso sólo vamos a considerar la última tarea que haya empezado, que es la que determinará el estado de  $c$  en  $\tau_1$ . Llamemos entonces  $\theta = [[e, f]]$  a esa última tarea. Al final de esta sección analizaremos qué hacer con las demás tareas que  $c$  ya realizó antes de  $\tau_1$ . Veamos los posibles escenarios en que podría estar  $c$  antes de  $\tau_1$ :

- Si  $\tau_1 \leq q_f$ , i.e.  $\theta$  estaba programada para terminar después de  $\tau_1$ ,  $\theta$  está incluida en el modelo y  $c$  se encuentra realizando  $\theta$  en  $\tau_1$ . En tal caso redefinimos:

$$A_{cr}^+(e_{beg}^c) = \{(e_{beg}^c, \theta)\}$$

y en el modelo PLE forzamos a que esa actividad sea utilizada, i.e.  $z_a = 1$ , con  $a = (e_{beg}^c, e)$ . De esta manera el recurso  $c$  será traspasado obligatoriamente a  $\theta$ . Será el evento  $f$  el encargado de traspasar  $c$  luego de  $\tau_1$ .

Por ejemplo este es el caso de  $c_6$  en la Figura 3.12.

- Si  $\lim_c^- < q_f < \tau_1$ , i.e.  $\theta$  terminó antes de  $\tau_1$ , pero después del límite de inclusión,  $\theta$  está incluida en el modelo y  $c$  está en *traspaso* durante  $\tau_1$ . Si  $c$  tenía programado un almuerzo luego de  $\theta$  debemos respetarlo. Luego  $c$  está en conexión o almorzando en  $\tau_1$ . Al igual que en el caso anterior, forzaremos a que  $c$  sea asignado a la tarea  $\theta$ :

$$A_{cr}^+(e_{beg}^c) = \{(e_{beg}^c, \theta)\}$$

reemplazando  $z_a = 1$  en el modelo. Además, las actividades de traspaso de  $c$  desde la tarea  $\theta$ , i.e.  $a \in A_{cr}^+(c, \theta)$ , serán o bien todas de conexión o bien todas de almuerzo, según el tipo de traspaso que ya comenzó antes de  $\tau_1$ , con lo cual de  $A_{cr}^+(c, \theta)$  sólo conservaremos las actividades del tipo correspondiente. La siguiente tarea de  $c$  empezará después de  $\tau_1$ , entonces no podemos forzar ninguna en particular, sino que será determinada por la solución del modelo.

Por ejemplo este es el caso de  $c_4$  y  $c_5$  en la Figura 3.12. En particular  $c_4$  realizaba una conexión después de su última tarea y estará disponible 9:33 (cuando la conexión finalice). En cambio  $c_5$  realizaba un almuerzo después de su última tarea, entonces estará disponible 9:53. En ambos casos el traspaso completa su duración mínima después de  $\tau_1$ , pero observar que si  $\theta_4$  finalizara 9:15,  $c_4$  estaría disponible en  $\tau_1$ .

- Por último, si  $q_f \leq \lim_c^-$ , i.e.  $\theta$  terminó antes del límite de inclusión de tareas,  $\theta$  no fue incluida en el modelo. En tal caso,  $c$  ya realizó el almuerzo o conexión antes de  $\tau_1$  y en  $\tau_1$  está disponible para ser utilizado en la estación donde terminó  $\theta$ , supongamos  $k$  (i.e.  $f \in E_{arr}^k$ ). En este caso, debemos recalculamos el conjunto de actividades iniciales de  $c$ ,  $A_{cr}^+(e_{beg}^c)$ , incluyendo ahora traspasos a tareas que empiecen en la estación actual de  $c$  (en vez de su base) y que puedan ocurrir a partir de  $\tau_1$ :

$$(e_{beg}^c, \theta) \in A_{cr}^+(e_{beg}^c) \Leftrightarrow \theta = [[e', f']], \quad e' \in E_{dep}^k \text{ y } \tau_1 \leq q_{e'} + d_{e'}$$

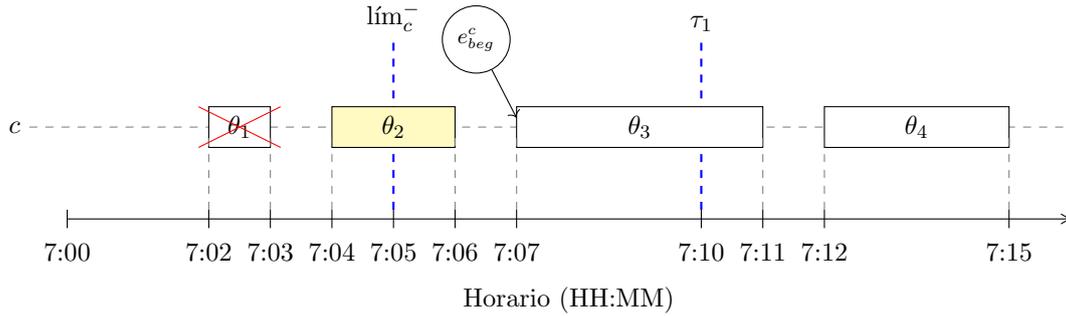
Como ya mencionamos previamente, este es el caso de  $c_1$  y  $c_2$  en la Figura 3.12.

En cuanto a las actividades de almuerzo, si  $c \in \mathbb{C}$  terminó su almuerzo antes de  $\tau_1$ , eliminamos todas sus actividades de almuerzo de la red. En cambio si  $c$  se encuentra en horario de almuerzo durante  $\tau_1$ , por las modificaciones que hicimos en  $N_c(\Theta)$ ,  $A_{cr}^+(c, \theta)$  tendrá únicamente actividades de almuerzo, siendo  $\theta$  la última tarea que realizada antes de  $\tau_1$ . Luego debemos eliminar todas las demás actividades de almuerzo de  $c$  del resto de las tareas.

Por último consideremos las tareas que fueron incluidas según los criterios anteriores, pero que ya fueron realizadas y terminaron antes de  $\tau_1$ . Por ejemplo, supongamos que una parte de la jornada original de  $c$  consiste en realizar las tareas  $[\theta_1, \theta_2, \theta_3, \theta_4]$  de manera secuencial. Supongamos que  $\lim_c^- = 7:05$  y  $\tau_1 = 7:10$  y que las tareas tienen estos horarios en el esquema original:

$\theta_1$ :	7:02 - 7:03
$\theta_2$ :	7:04 - 7:06
$\theta_3$ :	7:07 - 7:11
$\theta_4$ :	7:12 - 7:15

Con el criterio descrito anteriormente,  $\theta_1$  se elimina del modelo ya que  $q_f \leq \lim_c^-$  y las demás se mantienen. Las tareas  $\theta_2$  y  $\theta_3$  empezaron antes de  $\tau_1$  pero la última empezada por  $c$  fue  $\theta_3$ , con lo cual cae en el primer caso y se redefine  $A_{cr}^+(e_{beg}^c) = \{(e_{beg}^c, \theta_3)\}$ . Por último  $\theta_4$  será incluida en la reprogramación.



En este escenario,  $\theta_2$  se volvió inalcanzable desde  $e_{beg}^c$ . Si luego de haber actualizado  $N_{\mathbb{C}}(\Theta)$  sucede que  $\theta_2$  no es alcanzable desde ningún evento inicial  $e_{beg}^c$ , entonces  $\theta_2$  queda aislada de la red y por lo tanto podemos eliminarla (porque además ya fue realizada antes de  $\tau_1$ ). De hecho debemos eliminarla porque se volvería infactible una restricción de tipo (3.9).

Entonces eliminaremos todas aquellas tareas que queden en la situación descrita.

### Adaptaciones del modelo PLE

Si  $c \in \mathbb{C}$  terminó su almuerzo antes de  $\tau_1$ , eliminamos la restricción (3.14), ya que por las modificaciones de la sección anterior  $c$  no tiene actividades de almuerzo y entonces no será factible.

#### 3.6.2. Tripulaciones que deben ser reposicionadas

Cuando empezamos a construir el modelo lineal para tripulación en la Sección 3.4 nos basamos en una red  $N_{\mathbb{C}}(\Theta)$  que incluía a todas las tripulaciones y a todas las tareas. Luego de haber recortado la red con un límite inferior de tiempo sucedió que algunas tripulaciones se vieron afectadas, modificándose su conjunto de posibles

tareas iniciales. En particular, un posible escenario era que  $c$  haya completado una tarea antes de  $\tau_1$  y durante  $\tau_1$  estuviese en una estación  $k$  que no es su base. El modelo actual permite no utilizar a la tripulación en ninguna tarea (para contemplar el caso de tripulaciones de reserva), pero en este caso  $c$  debe ser reposicionada para regresar a su base.

Esto motiva la incorporación de un nuevo tipo de actividades de tripulación que nos permitan modelar la no utilización de una tripulación en la solución. Para cada  $c \in \mathbb{C}$  que en  $\tau_1$  se encuentra en su estación base  $b_c$ , agregaremos a  $A_{cr}$  la actividad  $a = (e_{beg}^c, e_{end}^c)$ . Es decir que  $z_a = 1$  representa explícitamente la decisión de no utilizar a  $c$  en ninguna tarea.

Para forzar a que todas las tripulaciones utilicen al menos una actividad, modificamos el conjunto de restricciones de capacidad (3.6) para que sea una igualdad en vez de una desigualdad, i.e.

$$\sum_{A_{cr}^+(e_{beg}^c)} z_a = 1$$

Dado que la sumatoria incluye las nuevas actividades, esta ecuación contempla las dos posibilidades: usar la tripulación en alguna tarea inicial y llevarla hasta el evento de finalización en su base, o bien si ya se encontraba en su base simplemente traspasar la tripulación al evento de finalización, sin participar de ninguna tarea.

Para mantener la factibilidad del modelo, este nuevo tipo de actividad no debe ser incluida en las restricciones (3.14) que deben forzar a tomar un almuerzo a una tripulación sólo si fue usada en alguna tarea. Tampoco debe agregarse una restricción del tipo (3.4) o (3.5) para estas actividades ya que  $x_f$  está indefinido para  $f \in E_{crew}$ . En todos estos casos alcanza con realizar los siguientes reemplazos en las definiciones:

- $A_{cr}^+(e_{beg}^c) \longrightarrow A_{cr}^+(e_{beg}^c) \setminus \{(e_{beg}^c, e_{end}^c)\}$
- $A_{cr}^-(e_{end}^c) \longrightarrow A_{cr}^-(e_{end}^c) \setminus \{(e_{beg}^c, e_{end}^c)\}$

### 3.6.3. Tareas posteriores al fin del bloqueo

El objetivo de esta sección es recortar parte de la red  $N_{\mathbb{C}}(\Theta)$  eliminando tareas que están programadas posteriormente a  $\tau_3$ , tiempo a partir del cual los eventos de tren no pueden ser demorados. Sin embargo, los servicios que empiezan antes de  $\tau_3$  podrían ser cancelados, incluso si algunos de sus eventos son posteriores a  $\tau_3$ . Más allá del método elegido para hacer este recorte, debemos *garantizar que las jornadas laborales y tareas que quedan fuera del modelo sean factibles*.

Un caso similar se presenta en Mosteiro [2] para el material rodante. El recorte de eventos posteriores a  $\tau_3$  implica agregar ciertas actividades de material rodante para asegurar que esté disponible en las estaciones correspondientes para poder continuar con la programación del día. El caso de la tripulación es distinto ya que no nos alcanza con garantizar que haya cierta cantidad en cada estación después de  $\tau_3$ , porque cada tripulación tiene un horario de fin y una estación base que debemos respetar. Luego el método de recorte que utilizemos debe mantener suficiente información que nos permita planificar el *futuro* de cada tripulación con muy poco nivel de detalle, pero garantizando factibilidad.

Realizaremos dos tipos de recorte. En primer lugar eliminaremos las tripulaciones  $c \in \mathbb{C}$  cuyas jornadas laborales originales no se vean afectadas por la reprogramación. Este tipo de recorte nos permitirá también eliminar las tareas que en la programación original sólo tenían asignada tripulación (de conducción o pasajera) que quedó fuera del modelo. Y en segundo lugar analizaremos cómo recortar algunas tareas posteriores a  $\tau_3$  realizadas en el cronograma original por alguna de las tripulaciones que sí fueron afectadas por la reprogramación.

### Tripulaciones que inician su jornada después del bloqueo

En principio deseamos que todas las tripulaciones cuyas jornadas laborales empiezan después de  $\tau_3$  se mantengan iguales. Entonces podríamos tomar como criterio de recorte que dichas tripulaciones se eliminen del modelo y de alguna manera garantizar la factibilidad. Sin embargo, hay casos donde este recorte genera una infactibilidad posterior a  $\tau_3$  imposible de resolver, como en el ejemplo de la Figura 3.13.

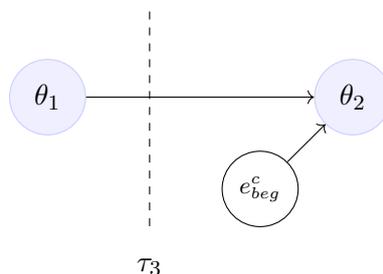


Fig. 3.13: Ejemplo de una tarea y una tripulación que no pueden ser eliminadas del modelo. Sea  $t = [\theta_1, \theta_2]$  un tren tal que  $\theta_1$  en el cronograma original ocurre antes de  $\tau_3$  y  $\theta_2$  después. Sea  $c$  una tripulación que comienza su jornada después de  $\tau_3$  y su primera tarea programada es  $\theta_2$ . Además ninguna otra tripulación tiene asignada  $\theta_2$  en su jornada original. Si eliminamos del modelo a  $c$ , luego también deberíamos eliminar a  $\theta_2$ . Pero como  $t$  empieza antes de  $\tau_3$  es cancelable. Si el modelo cancela  $t$ , luego  $\theta_2$  es cancelada y  $c$  no puede comenzar su jornada. Entonces tanto  $\theta_2$  como  $c$  deben considerarse en el modelo de reasignación de tripulación para evitar esta infactibilidad posterior a  $\tau_3$ .

El hecho de que algunas tareas puedan ser canceladas después de  $\tau_3$  genera este y otros problemas. Esto motiva definir un nuevo hito importante en el tiempo que llamaremos  $\tau_4$ , tal que todas las tareas que empiezan estrictamente después de ese punto, no son ni demorables ni cancelables. Pero primero definamos estos términos con más precisión.

**Def. 3.1.** Un servicio de tren  $t$  es *cancelable* si su primer evento de salida  $e \in E_{dep}$  cumple que  $\tau_1 \leq q_e \leq \tau_3$ .

**Def. 3.2.** Una tarea  $\theta$  es *cancelable* si  $t_\theta$  es cancelable.

Observar que las tareas no se pueden cancelar arbitrariamente ya que el modelo permite cancelar trenes, no tareas. Entonces una tarea cancelable es aquella incluida en un tren cancelable.

**Def. 3.3.** Una tarea  $\theta = [[e, f]]$  es *demorable* si  $q_e \leq \tau_3$  y  $q_f \geq \tau_1$ .

Intuitivamente, una tarea es demorable cuando agrupa uno o más eventos  $e'$  tales que  $q_{e'} \in [\tau_1, \tau_3]$ . Dicho de otra manera,  $\theta$  no es demorable si  $q_f < \tau_1$  o bien  $q_e > \tau_3$ , es decir todos los eventos que agrupa está fuera de los límites de reprogramación.

**Prop. 3.3.** Toda tarea  $\theta = [[e, f]] \in \Theta$  tal que  $q_e > \tau_4$  no es ni demorable ni cancelable, siendo

$$\tau_4 \stackrel{\text{def}}{=} \max(\{\tau_3\} \cup \{q_e \mid \theta = [[e, f]], \theta \in \Theta, \theta \text{ es cancelable}\})$$

*Demostración.* Sea  $\theta = [[e, f]]$  una tarea tal que  $q_e > \tau_4$ .

Por un lado, dado que  $\tau_4 \geq \tau_3$ ,  $q_e > \tau_3$ , entonces  $\theta$  no es demorable.

Por otro lado, supongamos que  $\theta$  es cancelable. Luego por la definición de  $\tau_4$ ,  $\tau_4 \geq q_e$ , pero partimos de  $q_e > \tau_4$ , entonces es un absurdo, que se produjo por suponer que  $\theta$  es cancelable. Luego no es cancelable.  $\square$

Observar que reemplazando  $\tau_3$  por  $\tau_4$  en el ejemplo de la Figura 3.13 el problema mencionado desaparece, ya que si  $\theta_2$  es posterior a  $\tau_4$  está garantizado que no es cancelable, i.e. que  $t$  no es cancelable.

Esto motiva un criterio para eliminar del modelo tripulaciones y tareas posteriores a  $\tau_4$  que nos conduzcan a soluciones factibles dentro y fuera del modelo. Vamos a eliminar a las tripulaciones  $c \in \mathbb{C}$  tales que  $w_c^{on} > \tau_4$ , y luego eliminaremos todas las tareas que sólo eran realizadas por tripulaciones eliminadas. Esto nos asegura que sin importar lo que suceda en la reprogramación, las jornadas originales que estamos dejando afuera del modelo contienen tareas que no serán canceladas ni reprogramadas.

Podría suceder que alguna  $c \in \mathbb{C}$  eliminada, i.e.  $w_c^{on} > \tau_4$ , tuviera en su jornada original una tarea de pasajero, que era conducida por una tripulación  $c' \in \mathbb{C}$  que sí fue incluida en el modelo. Esta tarea en principio no será eliminada y en la siguiente sección veremos cómo asegurar que alguna tripulación conduzca esa tarea.

Por último, observar que estamos incluyendo las tripulaciones con  $w_c^{on} = \tau_4$ , porque podría suceder que la primer tarea de  $c$  iniciaba en  $\tau_4$  exactamente y luego podría ser cancelable. Una alternativa a la aproximación que planteamos aquí podría ser evaluar una condición más específica para cada tripulación y cada tarea, pero queda para trabajo futuro ya que por el momento cumplimos el objetivo de recortar la red. Se podría evaluar si un recorte más cuidadoso de las tripulaciones reduce el tiempo total para resolver el problema, usando más tiempo en pre-procesamiento.

### 3.6.4. Compleción de jornadas

Sea  $c \in \mathbb{C}$  tal que  $w_c^{on} \leq \tau_4$  y  $w_c^{off} \geq \tau_1$ , es decir una tripulación que no fue eliminada del modelo, cuya jornada original se verá afectada por la reprogramación.

Las tareas de la jornada original de  $c$  que empiezan después de  $\tau_4$  no son demorables ni cancelables. Pero incluso puede haber tareas anteriores a  $\tau_4$  que tampoco sean demorables ni cancelables, que corresponden a los trenes que empezaron entre  $\tau_3$  y  $\tau_4$ . En principio nos gustaría eliminar todas estas tareas y garantizar de alguna manera que  $c$  pueda realizarlas, es decir que se encuentre en la estación adecuada en el momento justo antes de empezar estas últimas tareas. Esta aproximación tiene varios problemas. Por un lado, implicaría modificar el armado de la red de evento-actividad para codificar la situación entre  $\tau_3$  y  $\tau_4$  de cada tripulación. Por otro lado, si otras cancelaciones o demoras impiden que  $c$  llegue a tiempo a la estación adecuada, estaríamos en un caso de infactibilidad.

Esto motiva a buscar una solución más flexible para representar una **compleción de jornada**. Dada una tripulación  $c \in \mathbb{C}$ , definiremos una tarea especial que notaremos  $\varphi_c$  que agrupará las últimas tareas no cancelables ni demorables de  $c$ . Observar que  $\varphi_c$  podría no estar definida en el caso de que no exista ninguna de dichas tareas.

**Def. 3.4.** Sea  $c \in \mathbb{C}$  una tripulación cuya jornada original es la secuencia de tareas  $[\theta_1, \dots, \theta_n]$  y supongamos que existe un  $i$ , con  $\theta_i = [[e, f]]$ , tal que:

- $q_e > \tau_3$
- $\{\theta_i, \dots, \theta_n\}$  son tareas no demorables ni cancelables.
- $i = 1$  o bien  $\theta_{i-1}$  es demorable o cancelable.

Sea  $\theta_n = [[e', f']]$  definimos la *compleción de la jornada original* de  $c$  como una tarea ficticia  $\varphi_c$  que agrupa la secuencia  $[\theta_i, \dots, \theta_n]$ , i.e.

$$\varphi_c = [[e, f']]$$

**Prop. 3.4.**  $\varphi_c$  no es ni demorable ni cancelable.

*Demostración.* Esta propiedad se puede derivar directamente del hecho de que todas las tareas agrupadas no son ni demorables ni cancelables.  $\square$

Luego para cada  $\varphi_c$  podemos eliminar del modelo la secuencia de tareas que agrupa y reemplazarlas por la nueva tarea ficticia, i.e. agregar  $\varphi_c$  a  $\Theta$ . De esta manera reducimos la cantidad de tareas que el modelo lineal tendrá en cuenta y se reduce el espacio de soluciones factibles. Como desventaja tenemos que las tareas eliminadas no podrán ser reasignadas libremente, pero recordemos que nuestro objetivo es incorporar restricciones del problema de reasignación en el modelo de reprogramación de horarios para reducir las cancelaciones en la tercera fase del framework iterativo del Capítulo 1, donde se resolverá el problema completo de reasignación. Con lo cual utilizar  $\varphi_c$  es una manera de simplificar las restricciones de tripulación para poder incluirlas en la primera fase sin agregar toda la complejidad del problema de tripulación.

### Horario de almuerzo durante $\varphi_c$

Supongamos que  $c$  en su jornada original tenía programado el horario de almuerzo entre las tareas que se agruparon en  $\varphi_c$ . En tal caso, partiremos  $\varphi_c$  y agregaremos una única actividad de almuerzo entre ambas partes, para mantener el almuerzo en el mismo horario original. De esta manera no necesitamos realizar modificaciones a las restricciones de almuerzo, ya que el almuerzo realizado durante  $\varphi_c$  se convierte en una actividad de almuerzo como cualquier otra. Este modelo fuerza a que el almuerzo sea realizado durante  $\varphi_c$  para respetar la jornada original, pero más adelante veremos que esta idea se puede flexibilizar para permitir realizar una conexión sin almuerzo.

### Adaptaciones del modelo PLE

Observar que  $t_{\varphi_c}$  no está definido ya que  $\varphi_c$  agrupa tareas de posiblemente distintos trenes. Pero como  $\varphi_c$  no es ni demorable ni cancelable, vamos a hacer un abuso de notación y tomar como válida la siguiente identidad:

$$y_{t_{\varphi_c}} \stackrel{\text{def}}{=} 0 \quad (3.19)$$

De esta manera las restricciones (3.9) y (3.10) quedan bien definidas para  $\varphi_c \in \Theta$ , forzando la realización de  $\varphi_c$  por un lado (ya que no es cancelable) y permitiendo la cantidad máxima de tripulaciones como pasajeras de ser necesario. Para garantizar que  $\varphi_c$  no es demorable no es necesario hacer asunciones adicionales ya que si  $\varphi_c = [[e, f]]$ , luego por definición sabemos que  $q_e > \tau_3$  y entonces  $d_e = 0$ , y las restricciones (2.3) garantizan que  $e$  no sea demorado.

También quedan bien definidas las restricciones (3.11) que luego permitirá penalizar si  $c$  no realiza su compleción de jornada. Dado que  $\varphi_c$  agrupa  $k$  tareas, la penalización se multiplica por  $k$ , para ser consistente en penalizar a cada tarea que  $c$  realizaba originalmente y no realiza en la jornada alternativa.

Ninguna otra restricción debe modificarse dado que construimos  $\varphi_c$  como una tarea más.

### Alternativas de asignación de $\varphi_c$

Las tareas ficticias  $\varphi_c$  son tratadas como cualquier otra tarea durante el armado del grafo de alcance. Sin embargo al momento de armar  $N_{\mathbb{C}}(\Theta)$  podemos elegir una de dos alternativas: respetar la asignación original de  $\varphi_c$  o bien reasignarlas.

La primer alternativa es la más sencilla, ya que consiste en definir el conjunto  $A_{cr}^+(\varphi_c)$  durante el armado de  $N_c(\Theta)$  tal que sólo contenga una actividad de salida que conectará  $\varphi_c$  con  $e_{end}^c$ . Es decir:

$$A_{cr}^+(\varphi_c) = \{(\varphi_c, e_{end}^c)\}$$

De esta manera las restricciones de flujo (3.8) implicarán que sólo  $c$  podrá realizar  $\varphi_c$  en cualquier solución, y por otro lado la restricción (3.9) aplicada a  $\varphi_c$  fuerza a que sea realizada, con lo cual  $c$  no tendrá alternativa más que realizarla (caso contrario el problema no tendrá solución). Observar que las restricciones (3.5) no pueden generar infactibilidad ya que en el caso de  $\varphi_c$  son trivialmente válidas por tratarse de la jornada original de  $c$ .

La segunda alternativa consiste en permitir que una tripulación pueda utilizar la compleción de jornada de otra. En este caso tratamos a las compleciones  $\varphi_c$  como cualquier otra tarea para determinar  $A_{cr}^+(\varphi_c)$ , utilizando la definición de la red de la Sección 3.2.3. Además en los casos donde partimos  $\varphi_c$  en el recreo de almuerzo, debemos agregar las actividades de almuerzo de todas las tripulaciones que puedan realizar  $\varphi_c$  y también todas las de conexión, permitiendo así que una tripulación pueda tomar el almuerzo fuera de  $\varphi_c$ .

Esta opción es menos restrictiva que la primera y resolverá la mayoría de las situaciones de infactibilidad. Sin embargo, en secciones posteriores veremos cómo aprovechar aún más la reasignación de compleciones de jornadas  $\varphi_c$  relajando las condiciones de horario de fin y de terminar en la estación base.

### 3.7. Mejoras a la factibilidad

Como mencionamos en la Sección 3.6.3, nuestro objetivo es incorporar restricciones de tripulación para que la reprogramación del cronograma ferroviario sea más realista, y evitar infactibilidad en la fase de reasignación de tripulación del framework iterativo. Entonces queremos evitar que la fase de reprogramación sea infactible debido a la visión restringida que tenemos del recurso de tripulación.

En las siguientes secciones analizaremos distintos casos donde el problema puede no tener solución bajo las restricciones actuales y veremos cómo relajarlas para tener mayor posibilidad de que exista una solución factible.

#### 3.7.1. Reposicionamientos en taxi

Hasta el momento nuestro modelo sólo admite reposicionamientos utilizando la misma red ferroviaria, pero como mencionamos en la Sección 3.1, existe la posibilidad de hacer reposicionamientos por fuera de la red, que en la literatura suelen llamarse **reposicionamientos en taxi**. Estos reposicionamientos implican un costo operativo más alto, pero también evitan un caso de infactibilidad por no encontrar manera de devolver a una tripulación a su base.

En una red de evento-actividad que se basa en el flujo de los recursos, modelar este tipo de reposicionamientos implicaría a priori que cualquier tarea pueda traspasar la tripulación a tareas que empiezan en distintas estaciones, asociando un costo alto a utilizar esas actividades. Luego el grafo de alcance definido en la Sección 3.2.2 dejaría de usar la condición espacial para relacionar dos tareas, es decir que un traspaso de tripulación podría hacerse siempre que el tiempo de finalización de una y el de inicio de la otra lo permitan. Esto agregaría una cantidad bastante grande de aristas, dependiendo de la cantidad de estaciones. Por ejemplo la primera tarea que termina, podría pasarle tripulación a *todas* las que empiecen después, sin importar en que estación.

Además en Huisman [7] se observa que la duración de los reposicionamientos en taxi depende del tránsito, que los vuelve más impredecibles y podría generar demoras adicionales. Entonces estos reposicionamientos sólo se permiten al principio y/o final de la jornada.

Para evitar agregar demasiadas actividades y siguiendo la observación de Huisman [7], soportaremos parcialmente estos reposicionamientos, permitiendo que cada tripulación  $c$  pueda terminar su jornada laboral en una estación que no sea su base. Es decir, no modelaremos el reposicionamiento entre cualquier par de tareas sino que sólo entre tareas y el evento final de inventario de esa tripulación. Para esto realizaremos dos modificaciones a la red de tripulación  $N_C(\Theta)$ .

Por un lado, agregaremos la actividad  $(e_{beg}^c, e_{end}^c) \in A_{cr}(c)$  para todo  $c \in \mathbb{C}$ , es decir incluso las tripulaciones que se encuentran en una estación distinta de su base en  $\tau_1$  y por ende deben realizar algún movimiento en la red (ver Sección 3.6.2), ahora podrán realizar un reposicionamiento en taxi directo a su base sin realizar más tareas, usando esta nueva actividad.

Por otro lado, la condición para agregar una tarea como la última posible de una tripulación que vimos en la Sección 3.2.3 se relaja, sacando la condición de que deba ser una tarea que termina en la estación base de  $c$ .

Finalmente agregamos a la función objetivo el siguiente término a minimizar:

$$W_{\text{taxi}} \sum_{c \in \mathbb{C}} \sum_{a \in A_{cr}^-(e_{end}^c)} \mathbf{dist}(a) z_a$$

donde  $W_{\text{taxi}}$  es la penalización general de que una tripulación no termine en su base y la función  $\mathbf{dist}(a) \in \mathbb{R}_{\geq 0}$ , con  $a = (f, e_{end}^c) \in A_{cr}^-(e_{end}^c)$  representa el costo asociado de *distancia*, por ejemplo el precio del taxi, para mover a la tripulación desde la estación del evento  $f \in E_{\text{train}}$  a la estación base de  $c$ . En el caso de  $f = e_{beg}^c$ , representa la distancia desde la estación en la que se encuentra  $c$  en tiempo  $\tau_1$  a su base. Si la estación del evento  $f$  es la base de  $c$ ,  $\mathbf{dist}(a) = 0$ , i.e. si  $c$  termina la jornada en su base la arista utilizada no tiene costo.

De esta forma no todas las estaciones tienen el mismo impacto en la función objetivo. La idea es que terminar en una estación geográficamente cerca de la estación base sea menos costoso que terminar del otro lado del mapa de la red. La función de distancia se puede calcular por ejemplo a partir de una tabla con las distancias o costos asociados a viajes entre cada par de estaciones.

### 3.7.2. Horas extra

Otra de las razones por las cuales nuestro modelo podría no tener solución es la de respetar los horarios de finalización de la jornada laboral, i.e. podría volverse infactible la restricción (3.5). Esto sucede trivialmente por ejemplo cuando se produce un bloqueo total de un tramo de vías y una tripulación necesita atravesarlo para regresar a su base. Una opción para salvar este caso podría ser utilizar el reposicionamiento en taxi de la sección anterior, pero podría ser conveniente seguir operando otros servicios parciales y cuando el bloqueo se resuelve, volver a su base unos minutos u horas más tarde de su horario de fin  $w_c^{off}$ . Esto generará un costo adicional para pagar las **horas extra** a la tripulación.

Para modelar esto, por un lado debemos remover la restricción de tiempo en el armado de  $A_{cr}^-(e_{end}^c)$  que realizamos en la Sección 3.2.3, i.e. permitir que  $c$  finalice su jornada incluso con tareas  $\theta = [[e, f]]$  tales que  $q_f > w_c^{off}$ . Aquí se podría limitar la cantidad máxima de horas extras permitidas, pero para el análisis de este trabajo lo dejamos libre, es decir que cada tripulación puede terminar tan tarde como se quiera.

Por otro lado deberíamos remover las restricciones (3.5) ya que impiden que

la tripulación realice horas extras. Pero como mencionamos previamente, también vamos a necesitar penalizar la cantidad de horas extras realizadas, entonces en vez de remover las restricciones, vamos a introducir un conjunto de variables que van a medir la holgura de (3.5) cuando  $z_a = 1$ , es decir van a contar la cantidad de tiempo extra que  $c$  utiliza desde su horario de fin hasta que efectivamente termina su última tarea.

Sea  $\kappa_c \in \mathbb{Z}_{\geq 0}$  una variable que indica el tiempo adicional que trabajó  $c$  en su jornada, definido como:

$$\kappa_c \stackrel{\text{def}}{=} \max\{0; x_f - w_c^{\text{off}}\}$$

siendo  $\theta = [[e, f]]$  la última tarea que fue asignada a  $c$  en su jornada alternativa. Si  $x_f \leq w_c^{\text{off}}$ ,  $\kappa_c = 0$  ya que no realizó horas extras. En cambio si  $x_f > w_c^{\text{off}}$ ,  $\kappa_c = x_f - w_c^{\text{off}}$ , es decir el tiempo adicional que trabajó.

Luego debemos reemplazar todas las restricciones (3.5) por las siguientes:

$$w_c^{\text{off}} + \kappa_c - x_f + (1 - z_a)M_f^c \geq 0 \quad \forall c \in \mathbb{C}, a = (f, e_{\text{end}}^c) \in A_{\text{cr}}^-(e_{\text{end}}^c) \quad (3.20)$$

donde  $M_f^c \stackrel{\text{def}}{=} q_f + d_f - w_c^{\text{off}}$ , y agregamos el siguiente término a minimizar a la función objetivo (2.1):

$$W_{\text{EH}} \sum_{c \in \mathbb{C}} \kappa_c \quad (3.21)$$

siendo  $W_{\text{EH}}$  la penalización general al uso de horas extras.

**Prop. 3.5.** *Si reemplazamos (3.5) por (3.20) y agregamos (3.21) a la función objetivo de (2.1), entonces cualquier solución óptima del modelo PLE garantiza que  $\kappa_c = \max\{0; x_f - w_c^{\text{off}}\}$  siendo  $\theta = [[e, f]]$  la última tarea que fue asignada a  $c$  en su jornada alternativa.*

*Demostración.* Por un lado, al haber agregado las variables sumando a la función objetivo las estamos minimizando, es decir que cada  $\kappa_c$  tomará el menor valor posible en cualquier solución óptima.

Por otro lado, veamos qué sucede si  $z_a = 1$ , i.e. si la actividad es utilizada en la solución y por lo tanto  $f$  es el último evento que realiza  $c$ . Por (3.20) se cumple:

$$\kappa_c \geq x_f - w_c^{\text{off}}$$

Si  $x_f \leq w_c^{\text{off}}$ ,  $\kappa_c = 0$  (su valor mínimo). En cambio si  $x_f > w_c^{\text{off}}$ ,  $\kappa_c = x_f - w_c^{\text{off}}$  tomará el valor de la igualdad, que será el menor posible. Luego se cumple la igualdad deseada  $\kappa_c = \max\{0; x_f - w_c^{\text{off}}\}$ .

Por último, veamos que las restricciones se verifican si  $z_a = 0$ , independientemente del valor de las variables. Reemplazando en (3.20), debería verificarse:

$$x_f - w_c^{\text{off}} - \kappa_c \leq M_f^c$$

Siendo  $M_f^c = q_f + d_f - w_c^{\text{off}}$  y acotando las variables tenemos que:

$$\underbrace{x_f}_{\leq q_f + d_f} - w_c^{\text{off}} - \underbrace{\kappa_c}_{\leq 0} \leq q_f + d_f - w_c^{\text{off}} = M_f^c$$

Luego se verifica la desigualdad.  $\square$

### 3.7.3. Omitir almuerzo

Otra de las razones por las cuales una instancia del problema puede no ser factible es que no se pueda encontrar un tiempo de almuerzo para alguna tripulación  $c$  que cumpla con las restricciones que vimos en la Sección 3.4.3. Nuestro objetivo es obtener una reprogramación de los horarios de los trenes teniendo en cuenta restricciones de tripulación, pero ante una situación de infactibilidad, podemos relajar estas restricciones para poder llegar a una solución factible y luego usar el framework iterativo para mejorarla.

Entonces vamos a realizar modificaciones al modelo para permitir que una tripulación no tenga asignado un horario de almuerzo en su nueva jornada, con una penalización en la función objetivo para minimizar que esto suceda. En particular la única restricción que debemos modificar es (3.14), que fuerza a que todas las tripulaciones que fueron utilizadas en la solución deben tener un horario de almuerzo.

Sea  $y_c^{mb} \in \{0, 1\}$  la variable de cancelación de almuerzo, que análogamente a las variables de cancelación de trenes, vale 1 cuando el almuerzo de  $c$  es cancelado. Luego reemplazamos (3.14) por las siguientes restricciones:

$$y_c^{mb} + \sum_{a \in A_{mb}(c)} z_a = \sum_{a \in A_{cr}^+(e_{beg}^c)} z_a \quad \forall c \in \mathbb{C} \quad (3.22)$$

y agregamos a la función objetivo (2.1) el siguiente término a minimizar:

$$W_{mb} \sum_{c \in \mathbb{C}} y_c^{mb} \quad (3.23)$$

siendo  $W_{mb}$  la penalización por cancelar un almuerzo.

Observar que al sumar  $y_c^{mb}$  del lado izquierdo de (3.22) estamos permitiendo que no se tome ninguna arista de almuerzo con  $y_c^{mb} = 1$ , cuando la tripulación es utilizada en la solución.

Tener en cuenta que al realizar los recortes temporales a la red, hay tripulaciones que se incluyen en la reprogramación pero que ya tuvieron su horario de almuerzo antes de  $\tau_1$ . En esos casos habíamos eliminado todas las actividades de almuerzo y por consiguiente la restricción (3.14), y por ende tampoco vamos a incluir (3.22).

## 4. Resultados preliminares

En el presente capítulo realizamos un estudio experimental a fin de estimar la dificultad computacional y evaluar la calidad de la solución del problema *integrado* descrito en el Capítulo 3. Además compararemos este modelo con una variante *secuencial* que primero resuelve el problema presentado en Veulenturf et al. [1] y luego el problema integrado, teniendo en cuenta las demoras y cancelaciones del primer problema.

Todos los experimentos fueron realizados en un entorno con procesador Intel Core i5-6200U (4 cores a 2.3GHz), 8GB de RAM, sistema operativo Ubuntu 16.04 LTS de 64 bits, utilizando C++ y el paquete de optimización CPLEX versión 12.6.0.

### 4.1. Caso de estudio: Línea Mitre

Una de las problemáticas que enfrentamos a la hora de encarar nuestra experimentación fue la imposibilidad de disponer de *benchmarks* de acceso público para utilizar como datos de comparación.

En Mosteiro [2] se construyó una instancia a partir de datos públicos de la Línea Mitre. En la Figura 4.1 se observa la red geográfica de la Línea Mitre. Para simplificar los resultados preliminares, restringimos el análisis a un único ramal de esta red. El ramal escogido es el que une Retiro con Tigre, que es el mismo que se usó en Mosteiro [2]. Este ramal además es lineal (no hay intersecciones o bifurcaciones), lo cual añade una simplificación adicional. Nuestro trabajo a futuro incluirá el análisis de instancias más complejas, lo cual explicaremos con más detalle en el Capítulo 5.



Fig. 4.1: Línea Mitre - Mapa de recorridos

El ramal Retiro-Tigre cuenta con 17 estaciones. Cada una de ellas cuenta con dos vías y cada par de estaciones está unido por tramos de dos vías también. No disponemos del detalle de qué estaciones son de relevo para el problema de tripulación, pero asumiremos que tanto Retiro como Tigre deben ser de relevo (por ser cabeceras y el comienzo/fin de los servicios de tren) y en los experimentos incluiremos otras estaciones de relevo elegidas arbitrariamente, ya que de no haber otras, todas las tareas de conducción serían trenes completos, simplificando mucho el problema.

Para nuestros experimentos, tomamos el cronograma utilizado en días hábiles en período pre-pandemia COVID-19, el cual está compuesto de 162 servicios. Cada servicio demora 60 minutos desde una cabecera a la otra, y parten con una frecuencia de 14 minutos. La Figura 4.2 muestra un extracto del cronograma para los trenes

que viajan desde Retiro hacia Tigre<sup>1</sup>.

TREN N°	RETIRO	L. DE LA TORRE	BELGRANO C	NÚÑEZ	RIVADAVIA	YTE. LÓPEZ	OLIVOS	LA LUCILA	MARTINEZ	ACASSUSO	SAN ISIDRO	BECCAR	VICTORIA	VIRREYES	SAN FERNANDO	CARUPA	TIGRE
3001	05:00	05:11	05:14	05:18	05:21	05:23	05:26	05:29	05:31	05:33	05:35	05:38	05:42	05:44	05:47	05:49	05:54
3003	05:13	05:24	05:27	05:31	05:34	05:36	05:39	05:42	05:44	05:46	05:48	05:51	05:55	05:57	06:00	06:02	06:07
3005	05:26	05:37	05:40	05:44	05:47	05:49	05:52	05:55	05:57	05:59	06:01	06:04	06:08	06:10	06:13	06:15	06:20
3007	05:39	05:50	05:53	05:57	06:00	06:02	06:05	06:08	06:10	06:12	06:14	06:17	06:21	06:23	06:26	06:28	06:33
3009	05:52	06:03	06:06	06:10	06:13	06:15	06:18	06:21	06:23	06:25	06:27	06:30	06:34	06:36	06:39	06:41	06:46
3011	06:04	06:15	06:18	06:22	06:25	06:27	06:30	06:33	06:35	06:37	06:39	06:42	06:46	06:48	06:51	06:53	06:58
3013	06:17	06:27	06:30	06:34	06:37	06:39	06:42	06:45	06:47	06:49	06:51	06:54	06:58	07:00	07:03	07:05	07:10
3015	06:30	06:40	06:43	06:47	06:50	06:52	06:55	06:58	07:00	07:02	07:04	07:07	07:11	07:13	07:16	07:18	07:23
3017	06:43	06:53	06:56	07:00	07:03	07:05	07:08	07:11	07:13	07:15	07:17	07:20	07:24	07:26	07:29	07:31	07:35
3019	06:55	07:05	07:08	07:12	07:15	07:17	07:20	07:23	07:25	07:27	07:29	07:32	07:36	07:38	07:41	07:43	07:47
3021	07:08	07:18	07:21	07:25	07:28	07:30	07:33	07:36	07:38	07:40	07:42	07:45	07:49	07:51	07:54	07:56	08:01
3023	07:21	07:31	07:34	07:38	07:41	07:43	07:46	07:49	07:51	07:53	07:55	07:58	08:02	08:04	08:07	08:09	08:13
3025	07:34	07:44	07:47	07:51	07:54	07:56	07:59	08:02	08:04	08:06	08:08	08:11	08:15	08:17	08:20	08:22	08:27
3027	07:47	07:58	08:01	08:05	08:08	08:10	08:13	08:16	08:18	08:20	08:22	08:25	08:29	08:31	08:34	08:36	08:41

Fig. 4.2: Ramal Retiro-Tigre. Extracto del Cronograma 2016 para trenes viajando desde Retiro hacia Tigre, utilizado en días Lunes a Viernes hábiles.

En cuanto al material rodante, al igual que en el trabajo de Veelenturf et al. [1] y Mosteiro [2], sólo consideramos la cantidad mínima de unidades disponibles al inicio del día para garantizar la circulación de los trenes en operatoria normal. Es decir que no son consideradas unidades de repuesto. Esta cantidad mínima está determinada por 10 composiciones de material rodante en la estación cabecera Tigre. Esto es así dado que el cronograma indica que los trenes que parten desde Tigre comienzan su recorrido mucho antes (aprox. 1:15hrs), con lo cual alimentan con material rodante a la estación de Retiro una vez que terminan su recorrido.

En lo que respecta a la tripulación, debido a que no conseguimos datos reales de la Línea Mitre, creamos una asignación inicial respetando el convenio colectivo de trabajo del sindicato ferroviario<sup>2</sup>. Este convenio define jornadas laborales de 6 horas como máximo para el personal de conducción (que es la parte más importante de la tripulación y en la que basaremos nuestros experimentos), contando a partir de la primer tarea asignada, pero permite que se extienda “a los efectos de asegurar la continuidad de los servicios en los casos imprevistos y/o de fuerza mayor”. Esta norma es compatible con el modelo que presentamos en el capítulo 3 en el cual permitimos la realización de horas extra de ser necesario. La norma no lo aclara, pero asumiremos un tiempo mínimo de recreo de 45 minutos dentro de las 6hs de jornada, que se utilizará para almuerzo de ser necesario.

De manera análoga al material rodante, asumiremos que las tripulaciones tienen como estación base a Tigre y que luego van alimentando los trenes que parten desde Retiro, y no será necesario que ninguna tripulación tenga a Retiro como base. En el cronograma original, cada servicio tendrá asignada una única tripulación durante todo el recorrido y cada tripulación conducirá cuatro servicios completos, realizando un almuerzo de 60 minutos luego de los primeros dos, y terminando en su estación base luego del último. Este esquema de asignación determina 3 turnos de trabajo, mañana, tarde y noche. En las Figuras 4.3, 4.4 y 4.5 se observa la asignación inicial de tripulación para cada turno.

<sup>1</sup> El cronograma de la Figura 4.2 difiere sutilmente de los datos mencionados (y utilizados en nuestros experimentos) dado que el mismo corresponde al período 2016 y los datos utilizados corresponden al período 2015.

<sup>2</sup> Resolución 1634/2015 del Ministerio de Trabajo, Anexo I, Artículo 5. <http://servicios.infoleg.gob.ar/infolegInternet/anexos/255000-259999/258010/norma.htm>

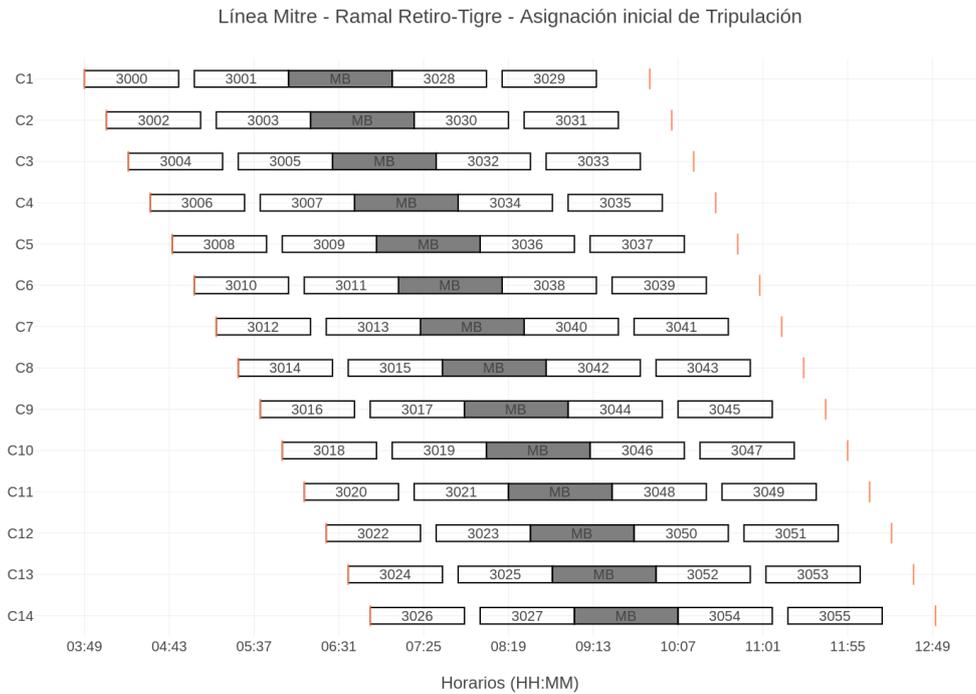


Fig. 4.3: Asignación de las tripulaciones del turno mañana. El eje horizontal representa el tiempo y el eje vertical contiene la jornada original de una tripulación por cada fila. Con líneas verticales de color naranja se indican los horarios de inicio y fin de las jornadas de 6hs. Las cajas sin relleno son las tareas y en el interior se indica el servicio de tren operado. Las cajas con relleno gris son los recreos de almuerzo (o descanso en el caso del turno mañana). Cada servicio tiene una única tarea ya que las únicas estaciones de relevo son Retiro y Tigre. Los servicios con número par operan de Tigre a Retiro. Los servicios con número impar operan de Retiro a Tigre.

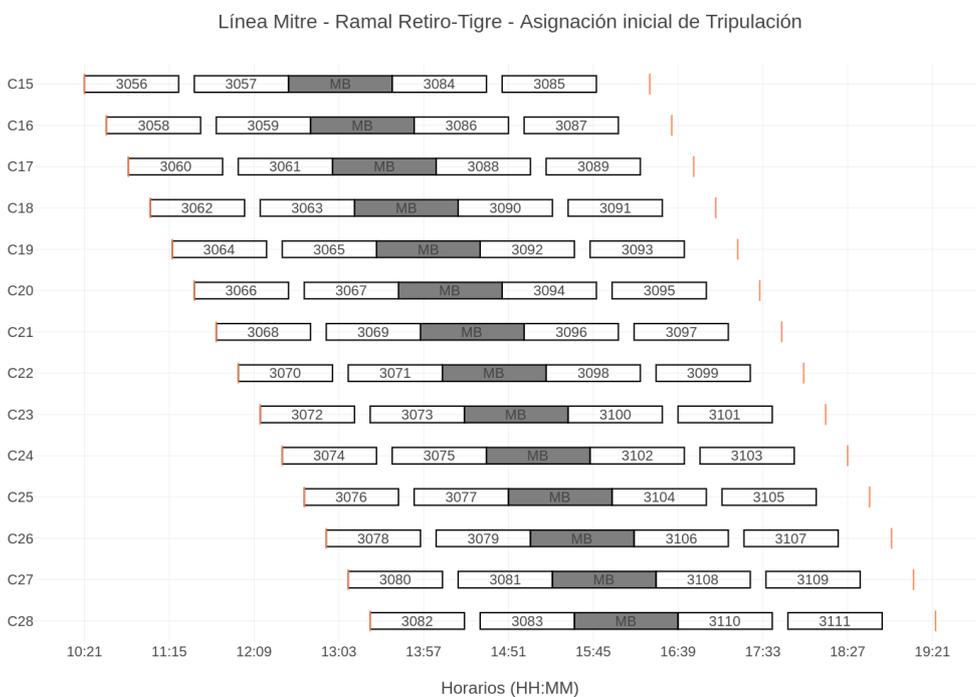


Fig. 4.4: Asignación de las tripulaciones del turno tarde.

#### 4.1.1. Ampliación a la cantidad de tareas por servicio

Como mencionamos previamente, en la asignación original de tripulación a los servicios, cada servicio se corresponde a una única tarea que es realizada por una

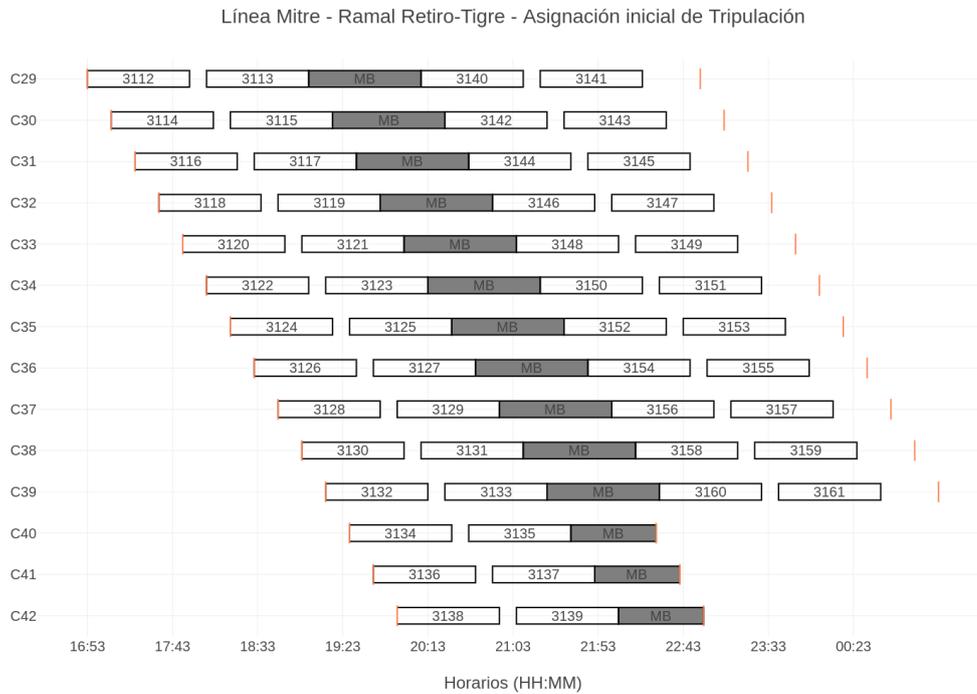


Fig. 4.5: Asignación de las tripulaciones del turno noche. Las jornadas de las tripulaciones 40, 41 y 42 son de 3hs únicamente y sólo operan dos servicios, entonces el resto del tiempo queda libre (50 minutos, que cumple el mínimo de 45 minutos de recreo).

única tripulación. Esto garantiza un cubrimiento de los servicios de manera análoga al cubrimiento por material rodante, donde todas las tripulaciones tienen como base a la estación Tigre.

Sin embargo, tener una única tarea por servicio resulta una simplificación demasiado fuerte del problema de tripulación, por lo que incluiremos una sutil modificación a la instancia original. Tomaremos a las estaciones Núñez y Martínez como estaciones de relevo de tripulación. Es decir que estas estaciones permitirán el cambio de una tripulación de un material rodante a otro (conexión) y permitirán tomar el recreo o almuerzo en las mismas. En la Figura 4.6 se muestran los horarios del turno mañana. Observar que la única diferencia con la Figura 4.3 es la subdivisión de cada tarea en 3 partes.

Estas estaciones se seleccionaron de manera arbitraria, pensando en subdividir a la red en tres partes de distintas longitudes (computadas como cantidad de tramos de vías recorridos): las tareas que unan el tramo Retiro-Núñez de longitud 3, Núñez-Martínez de longitud 5 y Martínez-Tigre de longitud 8.

La asignación original no se modifica, pero cuando introduzcamos un bloqueo, estas estaciones jugarán un rol importante en la posibilidad de que una tripulación por ejemplo termine su jornada en Núñez y otra pueda ejecutar servicios a partir de allí.

## 4.2. Parámetros

En cuanto a los parámetros relacionados con eventos, el horario original  $q_e$  de cada evento  $e$  queda determinado a partir del cronograma del ramal Retiro-Tigre, del cual incluimos un extracto en la Figura 4.2. La capacidad  $i_e$  de los eventos de inventario también queda determinada a partir de la descripción realizada en 4.1 de nuestra red.

Otro parámetro relativo a los eventos que debe ser definido es el mayor retraso permitido  $d_e$  para cada evento  $e$ , cuyo valor tiene una gran relevancia. Valores chi-

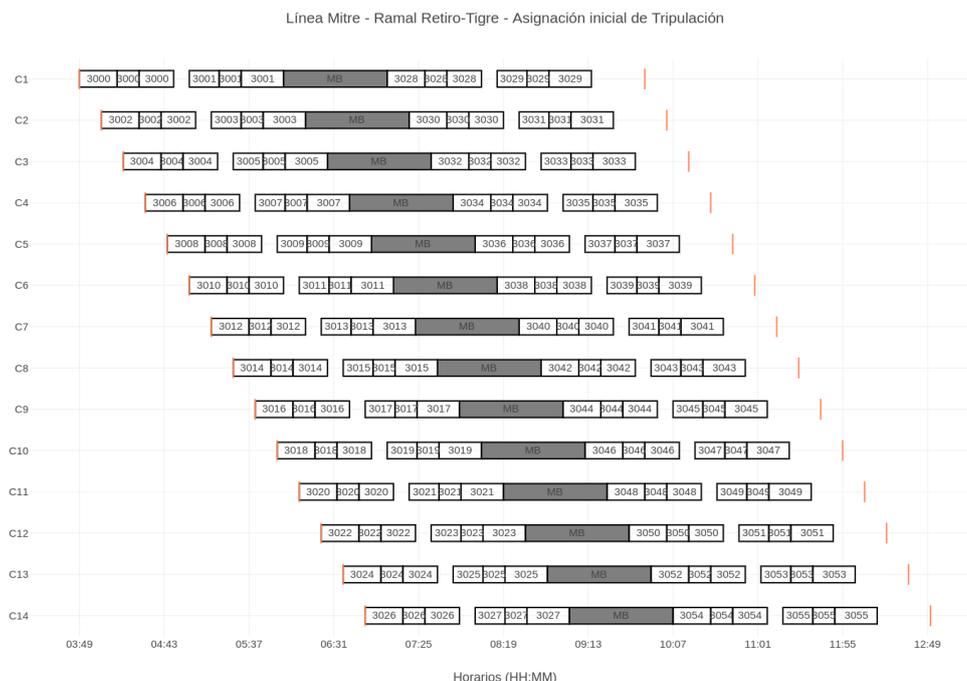


Fig. 4.6: Asignación de las tripulaciones del turno mañana con estaciones de relevo en Retiro, Núñez, Martínez y Tigre.

cos para  $d_e$  implican menos opciones de reprogramación, obligando a cancelar más trenes. Por otro lado, valores grandes dan lugar a muchas más opciones, generando problemas en principio más difíciles, pudiendo causar un fuerte impacto en los tiempos de cómputo. Para nuestros experimentos, utilizaremos distintos valores para  $d_e$ , a saber: 2, 3, 5, 7, 10, 12 y 15 minutos.

En lo que respecta al bloqueo, realizaremos experimentos con un bloqueo total (2 vías) entre Belgrano y Núñez. Esto implica que las tareas de manejo que unen Retiro con Núñez deberán partirse en dos, según el esquema que explicamos en la Sección 3.5. La duración del bloqueo determinará el recorte temporal de la red descrito en la Sección 3.6. Una duración corta implica un problema más chico, con menos servicios a considerar pero también más restrictivo. En cambio una duración larga implica un problema más grande con muchas más opciones a considerar y por lo tanto puede llevar a mucho tiempo de ejecución. Es por esto que experimentaremos con distintas duraciones de bloqueo, en saltos de media hora: 30, 60, 90, 120, 150, 180, 210 y 240 minutos. El tiempo de recuperación entre que finaliza el bloqueo en  $\tau_2$  hasta volver al funcionamiento normal en  $\tau_3$  lo fijaremos en 50 minutos.

En cuanto a los parámetros relativos a las actividades, utilizamos como tiempos mínimos de viaje de cada actividad (i.e.  $L_a$ ) a los tiempos de viaje del cronograma 4.2. Luego, para el nuevo cronograma estamos asumiendo que un tren no puede recorrer un tramo entre estaciones más rápidamente que en el programa original, ni tampoco forzamos que el tiempo de viaje tenga que ser mayor. Asimismo, en el cronograma 4.2 los arribos y posteriores salidas en una estación se encuentran programados en el mismo horario. Luego, todos los valores  $L_a$  relativos a actividades de espera en una estación en nuestro modelo serán 0, permitiendo el mismo comportamiento que en el programa original.

Los parámetros  $L_a$  de las actividades de recursos, es decir, los tiempos de seguridad, son valores que dependen de las características de la infraestructura de la red y de las restricciones operativas. Dichos datos no son de acceso público para nuestro caso de estudio. Luego, tomamos como valores para estos parámetros los mismos que se utilizan en Veelenturf et al. [1]. Para vías de tramos el tiempo de seguridad es de 2 minutos para trenes operando en una misma dirección y de 0 minutos en

dirección contraria. Para el traspaso de vías de estación, el tiempo de seguridad es de 2 minutos. En cuanto al material rodante, para las actividades de traspaso directo (sin pasar por el patio de maniobras) el  $L_a$  es de 5 minutos, mientras que para las actividades a través del patio de maniobras el tiempo de seguridad es de 10 minutos. Finalmente, para el caso de conexiones de tripulación usaremos también 5 minutos ya que en el cronograma original asumimos que la tripulación se mueve junto al material rodante, y para los tiempos de almuerzo usaremos 45 minutos, como mencionamos en la Sección 4.1.

La máxima cantidad de tripulaciones que viajan como pasajeras en una tarea  $\mathbb{P}_{\text{pas}}$  se tomó como 2, i.e. en una misma tarea podríamos tener hasta tres tripulaciones, una conduciendo y dos reposicionándose. Un valor de  $\mathbb{P}_{\text{pas}} = 0$  implicaría que no permitimos reposicionamientos dentro de la red, lo cual debería generar más cancelaciones, en cambio un valor grande ampliaría el espacio de soluciones, permitiendo reducir las cancelaciones, pero probablemente aumentando los tiempos de cómputo. Se dejará como trabajo futuro estudiar estos casos en concreto.

Como mencionamos en el Capítulo 3, el horario de almuerzo debe programarse no demasiado temprano en la jornada ni demasiado tarde. Para esto habíamos definido dos parámetros  $\tau_1^{mb}$  para el tiempo máximo desde que inicia la jornada laboral hasta que empieza el almuerzo y  $\tau_2^{mb}$  para el tiempo máximo desde que finaliza el almuerzo hasta que termina la jornada. Dado que las jornadas son de 6 horas y el almuerzo debe durar al menos 45 minutos, parece razonable tomar ambos valores en 3.5 horas. Es decir que cada tripulación deberá tener su almuerzo (o recreo) de 45 minutos antes de que se cumplan 3.5 horas de trabajo, y luego de terminar el almuerzo, no puede trabajar más de 3.5 horas antes de su horario de finalización ( $w_c^{off}$ ). Observar que las horas extra no se tienen en cuenta para esta restricción.

Por último, con la intención de comparar la calidad de las ejecuciones en un contexto de aplicación en tiempo real, vamos a poner un límite de 5 minutos para resolver el problema de optimización (excluyendo armado del modelo y lectura/escritura de datos). De esta manera, las métricas de calidad que analicemos estarán reflejando lo que se esperaría en una aplicación de tiempo real, y permitirá distinguir por ejemplo que en 5 minutos se llega a una muy buena solución, aunque el óptimo y la prueba de optimalidad se consigan mucho tiempo después.

## Parámetros de CPLEX

En general utilizamos los parámetros por defecto del solver, salvo que forzamos a que la ejecución se realice en un único thread y de forma determinística, para obtener resultados reproducibles. En ciertos casos particulares nos encontramos con inestabilidad numérica, que resolvimos agregando los parámetros `CPX_PARAM_NUMERICALEMPHASIS = CPX_ON` y `CPX_PARAM_STARTALG = CPX_ALG_BARRIER` (usar algoritmo Barrier para resolver la relajación inicial). Queda como trabajo a futuro experimentar con formas de recorrer el árbol de B&C y otros algoritmos para la relajación inicial.

### 4.3. Penalizaciones

En Mosteiro [2] se trabajó con dos penalidades:  $\lambda_t$  por cancelar el tren  $t$  y  $\mu_e$  por cada minuto que se demora el evento  $e$ . En particular, para definir  $\lambda_t$  se utilizó un parámetro  $W_\lambda$  multiplicado por la cantidad de minutos de duración de  $t$  en el cronograma original, y para definir  $\mu_e$ , un parámetro global  $\mu_e = W_\mu$ . Se realizaron experimentos con  $W_\mu = 1$  y  $W_\lambda \in \{1000, 1500, 3000\}$ .

En nuestro trabajo se introducen varias penalizaciones nuevas:  $W_\delta$  para los cam-

bios de las jornadas alternativas con respecto a las originales y  $W_p$  para las tareas de reposicionamiento como pasajeros (Sección 3.4.2),  $W_{\text{taxi}}$  para reposicionamiento en taxi (Sección 3.7.1),  $W_{\text{EH}}$  para horas extra (Sección 3.7.2) y por último  $W_{\text{mb}}$  para omitir almuerzos (Sección 3.7.3).

Una de las mayores dificultades a la hora de experimentar con el problema integrado es definir los valores de todas estas penalizaciones. Dado que cada una mide distintas cosas en distintas unidades, no hay una clara relación que permita definir una en función de otras. Para simplificar el análisis, vamos a dividir estos costos en tres categorías, según el impacto que tienen en el problema:

- Maximizar calidad del servicio: Consiste en minimizar demoras ( $W_\mu$ ) y cancelaciones ( $W_\lambda$ ).
- Minimizar problemas de recursos humanos: Consiste en minimizar las omisiones de almuerzos ( $W_{\text{mb}}$ ) y cambios en las jornadas ( $W_\delta$ ).
- Minimizar gastos de la empresa: Consiste en minimizar principalmente horas extra ( $W_{\text{EH}}$ ) y reposicionamientos ( $W_{\text{taxi}}$  y  $W_p$ ).

Como mencionamos a lo largo de este trabajo, el objetivo principal siempre es maximizar la calidad del servicio, con lo cual buscaremos principalmente minimizar las cancelaciones y en menor medida las demoras. En segundo lugar vamos a minimizar el uso de horas extra y viajes en taxi ya que implican un costo para la empresa, junto con las cancelaciones de almuerzos que tienen un impacto fuerte a nivel recursos humanos. Por último tendremos en cuenta la minimización de cambios y los reposicionamientos como pasajero en la red, ya que son los atributos de calidad menos críticos y utilizarlos en mayor medida puede ayudar mucho en mejorar la calidad del servicio.

Para nuestros experimentos, utilizaremos valores de penalizaciones que resultan razonables y generan resultados bastante variados en cuanto a la utilización de las características que se penalizan.

Para las penalizaciones por cancelaciones y demoras, utilizaremos  $W_\mu = 1$  y  $W_\lambda = 1500$ . Elegimos estos valores por un lado porque fueron utilizados en los experimentos del trabajo de Mosteiro [2] y por otro lado por el siguiente análisis. Un tren completo de la instancia Mitre tiene 32 eventos. Con demoras máximas de 15 minutos por evento (que es el máximo de demoras que experimentamos), demorar todos los eventos del tren tendría un costo de  $32 \times 15 \times W_\mu = 480$  y demorar todos los eventos de los 162 servicios tendría un costo de 77760. Por otro lado, un tren completo tiene una duración de 60 minutos, entonces cancelarlo completo tendría un costo de  $60 \times W_\lambda = 90000$ . Es decir, cancelar un único tren completo es más costoso que demorar al máximo todo el cronograma, que nunca va a suceder ya que sólo consideramos una ventana de tiempo alrededor del horario del bloqueo. Entonces estos valores nos aseguran que siempre se preferirá demorar antes que cancelar.

Teniendo en cuenta el análisis previo, tomaremos  $W_{\text{mb}} = 22500$ , que corresponde a penalizar con 500 cada minuto de almuerzo cancelado si consideramos que la duración de un almuerzo es de 45 minutos como mínimo. Esto significa que cancelar un minuto de tren es tres veces más costoso que uno de almuerzo.

En cuanto a las horas extra, tomaremos  $W_{\text{EH}} = 500$ , lo cual implica que 3 horas extra equivalen a cancelar un tren completo. En los resultados veremos que son pocos los casos donde se llega a utilizar más de 1 o 2 horas extra, con lo cual este costo resulta razonable para la instancia utilizada.

Para penalizar la distancia recorrida en taxi, por un lado definiremos la función  $\mathbf{dist}(a)$ ,  $a = (f, e_{\text{end}}^c) \in A_{\text{cr}}^-(e_{\text{end}}^c)$  como la cantidad de tramos de vía necesarios para llegar desde la estación del evento  $f$  hasta la base de  $c$ . Por ejemplo, la cantidad de tramos de vía entre Virreyes y Tigre es de 3 (Virreyes - San Fernando - Carupá -

Tigre), con lo cual una tripulación cuya base es Tigre pero termina su jornada alternativa en Virreyes, tendrá que recorrer una distancia  $\mathbf{dist}(a) = 3$ . El costo asociado a esta distancia será  $W_{\text{taxi}} = 500$ , lo cual, a grandes rasgos, implica que 11 tripulaciones podrían terminar su jornada en Retiro y regresar en taxi a Tigre para *salvar* un tren completo.

La cantidad de tripulaciones viajando como pasajeras en una tarea será penalizada con un valor de  $W_p = 1$  ya que no representan un costo tan elevado para la empresa y además está restringida la máxima cantidad de pasajeros por  $\mathbb{P}_{\text{pas}}$ .

Por último, utilizaremos un valor de  $W_\delta = 100$  para minimizar los cambios a la jornada laboral. Recordemos que las tareas canceladas no se penalizan como cambio en la jornada (para evitar la doble penalización). Esto implica que utilizar valores muy elevados de esta penalidad en casos donde no se llega al óptimo, un 90 % de los servicios se cancelan, ya que resulta mejor cancelar tareas que modificar jornadas.

## 4.4. Implementaciones integrada y secuencial

A fines de realizar un análisis cualitativo y de escalabilidad, comparamos dos implementaciones del modelo.

La implementación **integrada** construye el modelo lineal que combina las restricciones de reprogramación de horarios, reasignación de material rodante y reasignación de tripulación con todas las mejoras a la factibilidad mencionadas en secciones previas.

La implementación **secuencial** primero resuelve el problema de reprogramación de horarios con restricciones de material rodante (Veelenturf et al. [1]). Del resultado extrae los trenes que fueron cancelados ( $t$  tal que  $y_t = 1$ ) y los nuevos horarios de los eventos de todos los eventos de tren  $x_e$  (cancelados o no), y luego construye el problema que integra los tres tipos de restricciones (horarios, material rodante y tripulación), fijando dichas variables  $y_t$  y  $x_e$ .

Observar que la implementación secuencial se asemeja a la primer iteración del framework iterativo (Capítulo 1). Hay una primera fase donde se resuelve el problema de reasignación de horarios con algunas restricciones de material rodante. Considerando que la red que utilizamos para experimentar tampoco permite partir el material rodante o cambiarlo antes de finalizar el recorrido, podemos asumir que las restricciones de material rodante consideradas en la primera fase son suficientes y por ende la etapa de asignación de material rodante del framework no agrega cancelaciones. Entonces pasamos a resolver la reasignación de tripulación directamente como segunda fase, pero teniendo en cuenta las restricciones de material rodante nuevamente, fijando los horarios y cancelaciones de la fase previa. Además de permitir agregar cancelaciones, estamos agregando la posibilidad de modificar la asignación de material rodante.

### 4.4.1. Parámetros de ejecución

Además de los parámetros del modelo de la Sección 4.2, vamos a experimentar con distintos parámetros de ejecución que determinarán el uso de mejoras a la factibilidad para obtener la solución. Utilizaremos los siguientes parámetros:

- **BASE+ORIG:** Resolver el modelo construido en la Sección 3.4 con los agregados de las secciones 3.5 y 3.6, sin las mejoras a la factibilidad de la Sección 3.7 y forzando a que cada tripulación realice su propia compleción de jornada, como se explicó en las *Alternativas de asignación de  $\varphi_c$*  de la Sección 3.6.4.
- **BASE:** Igual que BASE+ORIG, pero utilizando la segunda alternativa de asignación de  $\varphi_c$ , que consiste en permitir que cualquier tripulación realice cual-

quier  $\varphi_c$ .

- TAXI: Igual que BASE, pero agregando la posibilidad de regresar en taxi a la base al final de la jornada, como explicamos en la mejora de factibilidad de la Sección 3.7.1.
- HE: Igual que BASE, pero agregando la posibilidad de terminar la jornada después de  $w_c^{off}$ , como explicamos en la mejora de factibilidad de la Sección 3.7.2.
- CMB: Igual que BASE, pero agregando la posibilidad de cancelar almuerzos, como explicamos en la mejora de factibilidad de la Sección 3.7.3.
- TAXI+HE+CMB: Igual que BASE, pero agregando las tres mejoras de factibilidad.

Observar que no realizamos combinaciones de BASE+ORIG con las mejoras de factibilidad, ya que como veremos en los resultados, BASE+ORIG tiene un porcentaje de factibilidad muy bajo que se mejora mucho con BASE, entonces no tenía mucho sentido intentar mejorarlo con las otras alternativas.

## 4.5. Métricas de calidad

Teniendo tantos objetivos a optimizar, debemos optar por aquellos más relevantes a la hora de comparar soluciones y ante soluciones similares en uno de los objetivos, pasar a observar los demás. A continuación introducimos algunas métricas de calidad que luego serán utilizadas para presentar los resultados.

Como mencionamos en la Sección 4.3, la calidad del servicio será nuestra métrica de calidad más importante. En particular, calcularemos el porcentaje de cancelaciones realizado por una solución. Para esto, sumaremos por un lado la duración en minutos de cada uno de los servicios cancelables (según las definiciones del Capítulo 2) y por otro la duración en minutos de los servicios que efectivamente se cancelaron. Esto nos permite comparar soluciones independientemente de la cantidad de trenes que son incluidos en la reprogramación, por ejemplo al variar la duración del bloqueo ( $\tau_2 - \tau_1$ ).

Para medir las demoras, tomar el promedio o algún percentil tiene la contra de que muchas veces valen cero, ya que es común que muchos eventos no sean demorados. Además los eventos de trenes cancelados tienen demora nula (por restricciones (2.5)) entonces debemos asegurarnos que no alteren la medición de demoras. Entonces para simplificar tomaremos la suma de demoras de todos los eventos en la red, lo que equivale a sumar las demoras de los eventos de trenes no cancelados (ya que los cancelados suman cero).

En cuanto a la reasignación de tareas y cambios en jornadas laborales, calcularemos el porcentaje de tareas reasignadas, utilizando las variables  $\delta_{\theta,c}$  definidas en la Sección 3.4.2. Para cada tripulación  $c$  y para todas aquellas tareas  $\theta$  en su jornada original para las cuales está definida una variable  $\delta_{\theta,c}$  (inclusive las compleciones de jornada  $\varphi_c$ ), se calcula el porcentaje de cuántas fueron efectivamente reasignadas. En el caso de la instancia del tren Mitre todas las tareas tienen una única tripulación asignada en la asignación original, pero si tuviéramos una tarea asignada a dos tripulaciones por ejemplo, esta métrica distingue si la tarea fue reasignada para ninguna, una o ambas tripulaciones.

El resto de los objetivos se pueden medir sin cálculos adicionales: la cantidad de almuerzos omitidos, la cantidad de minutos extra trabajados, la distancia recorrida en taxi y la cantidad de tareas de reposicionamiento dentro de la red (tareas de pasajero).

## 4.6. Análisis de resultados

A continuación analizamos los resultados de ejecución de las implementaciones integrada y secuencial sobre las instancias generadas con los parámetros de la Sección 4.2. Es decir que vamos a tener un total de 56 instancias, correspondientes a todas las combinaciones de:

- Máxima demora ( $d_e$ ): 2, 3, 5, 7, 10, 12 y 15 minutos.
- Duración de bloqueo ( $\tau_2 - \tau_1$ ): 30, 60, 90, 120, 150, 180, 210 y 240 minutos.

### 4.6.1. Análisis de factibilidad

(a) Porcentaje de instancias								
Parámetros	Integrado				Secuencial			
	Sol	POpt	NFact	TO	Sol	POpt	NFact	TO
BASE+ORIG	50 %	32 %	50 %		45 %	45 %	55 %	
BASE	88 %	66 %	12 %		82 %	82 %	18 %	
TAXI	100 %	70 %			95 %	95 %	5 %	
HE	*100 %	70 %			100 %	100 %		
CMB	88 %	70 %	12 %		82 %	82 %	18 %	
TAXI+HE+CMB	96 %	70 %		4 %	100 %	100 %		

(b) No factibles y timeouts					
Parámetros	$d_e$	Integrado		Secuencial	
		NFact	TO	NFact	TO
BASE+ORIG	2,3,5,7	30,60,90,240		30,60,90,240	
	10,12,15	30,60,90,240		30,60,90,210,240	
BASE	2,3,5,7		240		240
	10,12,15		240		210,240
TAXI	10,12,15				210
HE	-		-		-
CMB	2,3,5,7		240		240
	10,12,15		240		210,240
TAXI+HE+CMB	15		150,240		

(c) Secuencial: Primer ejecución de casos numéricamente inestables			
Parámetros	$d_e$	$(\tau_2 - \tau_1)$	% Cancel.
TAXI	10	120	77.17 %
	15	120	Infact.
HE	5	180	92.95 %

Tab. 4.1: Soluciones obtenidas de las 6 combinaciones de parámetros sobre el total de las instancias del tren MITRE (56 instancias) ejecutando con un límite de 5 minutos. Todas las instancias para las cuales se encontró solución factible, la misma tuvo un porcentaje de cancelaciones por debajo del 25 %, salvo en \* que el caso  $d_e = 15$  con duración de bloqueo de 240 minutos obtuvo un 38.52 % con un GAP de 73.59 %. Los casos numéricamente inestables se volvieron a ejecutar con parámetros de estabilidad numérica antes de incluirlos en la primera tabla.

En la Tabla 4.1(a) comparamos la factibilidad de las implementaciones secuencial e integrada en función de los distintos parámetros de ejecución que presentamos en la Sección 4.4.1. Tomamos las siguientes métricas para analizar la factibilidad:

Sol Porcentaje de instancias para las que se obtuvo una solución factible.

POpt Porcentaje de instancias para las cuales se obtuvo prueba de optimalidad, i.e. el GAP fue nulo. La columna POpt del integrado y del secuencial no son comparables, ya que el óptimo global del secuencial es un óptimo local del integrado.

NFact Porcentaje de instancias que se determinaron no factibles.

TO (Timeout) Porcentaje de instancias que no encontraron solución factible antes del tiempo límite (5 minutos).

En la Tabla 4.1(b) se muestran las instancias que fueron no factibles o que no encontraron solución en el tiempo límite. Cada instancia está identificada por la máxima demora permitida ( $d_e$ ) y la duración del bloqueo ( $\tau_2 - \tau_1$ ). Por ejemplo en la primera fila de la tabla, con los parámetros BASE+ORIG las instancias correspondientes a todas las combinaciones de máxima demora de 2, 3, 5 y 7, y duraciones de bloqueo de 30, 60, 90 y 240, fueron determinadas no factibles dentro del tiempo límite de ejecución por ambas implementaciones integrada y secuencial.

Finalmente, en la Tabla 4.1(c) se mencionan tres casos donde el secuencial mostró inestabilidad numérica. Para resolverlos se agregaron los parámetros de CPLEX que mencionamos en la Sección 4.2 para mejorar la estabilidad y eso permitió resolverlos e incluirlos en las primeras tablas y en los análisis de calidad.

Analizando las tablas 4.1 (a) y (b) se observa el importante salto de factibilidad entre BASE+ORIG y BASE (permitiendo intercambios de compleciones de jornadas) tanto en el integrado como el secuencial. Forzar a retomar la jornada original después de  $\tau_3$  vuelve infactible la mitad de las instancias, mientras que permitir intercambiar las compleciones de jornadas entre tripulaciones mejoró la factibilidad cerca de un 40 %. Este porcentaje corresponde a bloqueos de duración corta, de 30, 60 y 90 minutos. Dado que el armado de las compleciones de jornada depende de  $\tau_3$  (ver Def. 3.4) y dado que estamos usando un tiempo fijo para ( $\tau_3 - \tau_2$ ), las tareas incluidas en cada  $\varphi_c$  dependen de la duración del bloqueo. Cuanto menos dure el bloqueo, más tareas serán incluidas en  $\varphi_c$ . Además el intervalo de reasignación de tareas es más corto y esto lleva a que en estos casos no sea posible disponer de  $c$  en la estación y tiempo adecuados para realizar  $\varphi_c$ , y por lo tanto la instancia se vuelva infactible. Esto muestra la necesidad de permitir intercambios de compleciones de jornada con una penalidad (como hicimos en BASE) y dada la gran diferencia en factibilidad, no profundizaremos el análisis de BASE+ORIG.

Las demás combinaciones de parámetros que introdujimos en la Sección 3.7 para mejorar la factibilidad, efectivamente logran resolver casi todas las instancias de prueba. En el caso de TAXI y HE, se resolvieron casi todas las instancias tanto en el integrado como el secuencial. En cambio en el caso de CMB hay un porcentaje de instancias que no se pudo resolver. Por último, la combinación de los tres parámetros (TAXI+HE+CMB) llegó al 100 % de instancias factibles en ambos casos, salvo que el integrado no logró encontrar solución factible en un 4 % en el tiempo fijado de 5 minutos, que corresponden a las dos instancias que se observan en la tabla (b). En una segunda ejecución se observó que esas dos instancias llegan a una solución factible antes de los 10 minutos.

Observar que en general las instancias más *difíciles* de resolver (en cuanto a factibilidad y tiempo) son aquellas con la duración de bloqueo más larga, de 240 minutos (i.e., 4 hs). En particular cuando observamos individualmente las mejoras de factibilidad, TAXI y HE pudieron resolverlas (tanto en integrado como secuencial), pero CMB no logró resolver ninguna, con ningún algoritmo ni máxima demora. Una hipótesis que tenemos en cuanto a la razón de este comportamiento es que el bloqueo, al ser total, divide a la red en 3 segmentos: Retiro-Belgrano, Belgrano-Núñez, Núñez-Tigre, y se cancelan los servicios entre Belgrano y Núñez. Dado que la base de tripulación se encuentra en Tigre, puede suceder que durante el bloqueo, una tripulación que quedó del lado Retiro-Belgrano llega al horario de fin de su jornada  $w_c^{off}$  y no tiene manera de regresar a Tigre a tiempo porque aún continúa el bloqueo. Como no permitimos viajes en taxi ni horas extra, la instancia se vuelve infactible.

Si comparamos los porcentajes de soluciones obtenidas entre el integrado y secuencial, en principio vemos que son similares, pero el integrado logra resolver instancias que el secuencial determinó como no factibles. Cabe aclarar que la infactibilidad siempre se produce en la segunda fase y no en la primera, i.e. la fase de reprogramación de horarios y reasignación de material rodante es factible en todas las instancias. El secuencial encuentra no factibles las mismas instancias que el integrado y además algunas instancias de duración de bloqueo de 210 minutos y distintos valores de  $d_e$ . Estos casos posiblemente se deban a que en la primera fase, el secuencial cancela o demora algún tren que es necesario para la factibilidad de la fase 2, pero el integrado los logra resolver ya que tiene en cuenta todas las restricciones al mismo tiempo. Este comportamiento podría depender (y es probable que así sea) de cómo CPLEX recorre el árbol de B&C, con lo cual queda como trabajo a futuro explorar con distintos parámetros del solver.

En la tabla 4.1 (a) también se observa el porcentaje de instancias para las cuales se obtuvo prueba de optimalidad. Por un lado, el secuencial siempre obtiene prueba de optimalidad cuando llega a una solución factible, y en análisis posteriores veremos que esto se condice con el tiempo de ejecución particularmente corto. Por otro lado, el integrado se ve afectado por la limitación de tiempo y en muchos casos no logra probar optimalidad. Sin embargo, en todas las soluciones se obtuvo menos de un 25 % de cancelaciones (salvo el caso indicado), lo cual nos da una idea de que las soluciones son razonables en cuanto a calidad, a pesar de no haber probado optimalidad.

#### 4.6.2. Análisis de calidad

##### Calidad sin mejoras de factibilidad (parámetros BASE)

En la Tabla 4.2 realizamos un estudio de las métricas de calidad presentadas en la Sección 4.5 en función de la demora máxima permitida  $d_e$ , sin utilizar parámetros que mejoran factibilidad (i.e. los parámetros BASE).

En cada fila agrupamos los resultados de 6 instancias, correspondientes a duraciones de bloqueos de 3hs o menos (30, 60, 90, 120, 150, 180 minutos). La elección de estas instancias es para garantizar que todos los casos analizados hayan encontrado una solución factible tanto en el integrado como el secuencial. Dado que estamos agrupando resultados, por cada métrica de calidad presentamos el promedio para las 6 instancias, pero también agregamos el valor mínimo y/o máximo para dar una idea de mejor/peor caso.

Las métricas analizadas son:

- % Cancelaciones: Porcentaje de tiempo de servicios cancelados en la solución sobre el total de tiempo de servicios cancelables.
- Demoras: Suma de las demoras (minutos) de los eventos de trenes no cancelados.
- Reposicionamiento: Suma de minutos utilizados por tripulaciones siendo reposicionadas dentro de la red (tareas como pasajeros).
- % Reasignaciones: El porcentaje de tareas efectivamente reasignadas sobre el total de tareas que era posible reasignar.
- % GAP: El GAP que reporta CPLEX, es decir la distancia entre la solución obtenida y la mejor cota inferior cuando no se pudo probar optimalidad.
- Tiempo: Segundos de ejecución del B&C y preprocesamiento de la instancia.

Las celdas vacías representan valores nulos.

Integrado													
$d_e$	% Cancelaciones			Demoras (min)		Reposic. (min)		% Reasig.		% GAP		Tiempo (seg)	
	min	prom	max	prom	max	prom	max	prom	max	prom	max	prom	max
2	18.66	19.41	20.25	9	56	80	111	40.0	55.34			7.04	21.49
3	15.95	18.96	20.25	17	56	80	130	37.62	54.37			7.54	22.34
5	14.01	15.49	17.72	83	120	106	158	41.12	47.58			6.95	12.84
7	14.01	15.49	17.72	81	120	100	139	41.12	47.58			12.75	32.03
10	12.21	13.6	15.28	1005	2304	97	130	42.29	50.49	4.18	12.96	124.81	t.l.
12	12.21	13.6	15.28	1029	2332	97	130	41.51	50.49	4.18	13.0	132.9	t.l.
15	14.01	16.47	23.35	814	2791	102	158	42.65	47.19	19.08	52.08	203.53	t.l.

Secuencial													
$d_e$	% Cancelaciones			Demoras (min)		Reposic. (min)		% Reasig.		% GAP		Tiempo (seg)	
	min	prom	max	prom	max	prom	max	prom	max	prom	max	prom	max
2	18.66	19.76	21.91	9	56	91	130	37.63	43.33			4.79	8.87
3	15.95	19.31	21.91	17	56	83	158	36.87	42.74			5.12	9.55
5	15.95	19.21	21.32	36	112	91	158	37.74	44.35			4.74	9.22
7	15.95	19.21	21.32	36	112	91	181	36.92	42.74			4.54	8.65
10	14.5	17.88	21.01	782	2184	91	158	37.9	44.35			6.48	14.19
12	14.5	17.03	20.25	798	2184	91	158	33.9	44.35			7.35	19.11
15	14.5	17.03	20.25	798	2184	88	158	34.26	44.66			9.44	23.65

Tab. 4.2: Métricas de calidad de integrado y secuencial con parámetros BASE (sin mejoras a la factibilidad) sobre la instancia MITRE, con duraciones de bloqueo de 3hs o menos (30, 60, 90, 120, 150, 180 minutos), i.e. cada fila agrupa las 6 instancias correspondientes al valor de  $d_e$  indicado. En todos los casos se encontró solución en el tiempo límite de 5 minutos.

Analizando la tabla observamos que en este subconjunto de instancias el integrado siempre mejora el porcentaje de cancelaciones en promedio (de las 6 instancias que se agrupan por fila). También se reduce considerablemente el mínimo para  $d_e \geq 5$  y el máximo en todos salvo  $d_e = 15$ , por falta de tiempo de ejecución. En general se observa que para  $d_e$  chicos, la diferencia no es muy grande entre secuencial e integrado, posiblemente porque hay poco margen para demorar trenes y entonces se terminan cancelando. En el resto de los casos, el secuencial asigna horarios y cancela trenes en la primera fase que luego llevan a cancelar más trenes en la segunda fase, posiblemente por no poder asignarles tripulación. En cambio el integrado cancela menos porque tiene en cuenta todas las restricciones al mismo tiempo y elige mejor que trenes cancelar.

Las demás métricas en general son mejores en el caso del secuencial. En particular el promedio de demoras depende de la cantidad de servicios que se operan. Al haber un mayor porcentaje de cancelaciones, menos servicios son operados y por lo tanto el total de demoras será menor. Lo mismo sucede con el porcentaje de reasignación de tareas, ya que las tareas de servicios cancelados no se reasignan y el porcentaje baja. Por último, en un nivel más general, los pesos utilizados en la función objetivo buscan reducir cancelaciones, con lo cual a menor porcentaje de cancelaciones es esperable que las demás métricas empeoren en general.

En cuanto a los tiempos de ejecución, observar que para los  $d_e$  más grandes, el integrado llega al límite de tiempo de 5 minutos, y cuanto más grande es  $d_e$ , mayor es el GAP entre la solución obtenida y la mejor cota inferior que obtuvo CPLEX, es decir que se encuentra más lejos de probar optimalidad. En cambio el secuencial nunca tardó más de 24 segundos en encontrar la solución y probar optimalidad. Esto naturalmente ocurre porque la separación en dos fases hace que el espacio de soluciones esté reducido y en la segunda fase en particular hay muchas variables fijas.

## Calidad con mejoras de factibilidad

En la Tabla 4.3 estudiamos las métricas de calidad presentadas en la Sección 4.5 en función de los distintos parámetros de ejecución de la Sección 4.4.1. En cada fila agrupamos los resultados de 35 instancias (listadas en la parte inferior) que son aquellas en las cuales obtuvimos prueba de optimalidad en todos los casos. Esta elección permite que las métricas sean comparables en igualdad de condiciones, eliminando la posibilidad que haya una solución mejor que por cuestión de tiempo no logramos encontrar. Las métricas analizadas son:

- % Cancelaciones: Porcentaje de tiempo de servicios cancelados en la solución sobre el total de tiempo de servicios cancelables.
- Demoras: Suma de las demoras (minutos) de los eventos de trenes no cancelados.
- % Reasignaciones: El porcentaje de tareas efectivamente reasignadas sobre el total de tareas que era posible reasignar.
- Reposicionamiento: Suma de minutos utilizados por tripulaciones siendo reposicionadas dentro de la red (tareas como pasajeros).
- Taxi: Suma de distancias recorridas por fuera de la red de cada tripulación, medida en cantidad de estaciones consecutivas.
- HE: Suma de minutos extra realizados por fuera del horario laboral de cada tripulación.
- CMB: Cantidad de tripulaciones a las que les fue cancelado el almuerzo.
- Tiempo: Segundos de ejecución del B&C y preprocesamiento de la instancia.

Dada la cantidad de instancias que se agrupan por fila, además del promedio, en algunos casos mostramos el máximo y/o mínimo para visibilizar los casos borde, y la mediana (p50) para dar una idea más clara de la distribución de los valores de la métrica.

Las celdas vacías representan valores nulos.

Integrado																			
Parámetros	% Cancelaciones			Demoras prom	% Reasig. prom	Reposic. (min)		Taxi (dist)			HE (min)			CMB (cant)			Tiempo (seg)		
	min	prom	max			prom	max	prom	p50	max	prom	p50	max	prom	p50	max	prom	p50	max
BASE	14.01	<b>16.88</b>	20.25	51	39.62	89	158										10.44	4.43	51.56
TAXI	14.01	<b>16.73</b>	20.25	55	40.25	82	148	<b>1.94</b>	26.0								14.3	9.12	85.44
HE	8.53	<b>14.41</b>	19.76	137	44.62	90	171			<b>12.74</b>	8.0	38.0					27.03	10.61	227.86
CMB	8.53	13.25	17.72	102	39.98	95	186						<b>1.54</b>	1.0	4.0		13.27	4.9	167.3
TAXI+HE+CMB	8.53	<b>13.06</b>	17.72	127	46.1	92	186	<b>2.6</b>	26.0	<b>9.49</b>	8.0	38.0		<b>0.89</b>	4.0		25.73	8.37	253.67

Secuencial																			
Parámetros	% Cancelaciones			Demoras prom	% Reasig. prom	Reposic. (min)		Taxi (dist)			HE (min)			CMB (cant)			Tiempo (seg)		
	min	prom	max			prom	max	prom	p50	max	prom	p50	max	prom	p50	max	prom	p50	max
BASE	15.95	19.37	21.91	70	35.11	82	181										4.95	3.09	14.17
TAXI	15.95	19.06	22.64	70	37.2	78	158	2.6	26.0								5.09	3.16	14.01
HE	8.53	14.92	19.76	70	44.24	91	209			24.11	16.0	138.0					5.56	3.8	14.82
CMB	8.53	<b>13.13</b>	17.72	70	39.22	90	190						2.57	<b>3.0</b>	4.0		4.3	2.84	12.73
TAXI+HE+CMB	8.53	13.13	17.72	70	46.58	84	190	2.83	21.0	11.66	8.0	38.0	1.34	<b>1.0</b>	4.0		5.15	4.03	13.63

Instancias agrupadas por fila (con prueba de optimalidad en todos los casos)

$d_e$	$(\tau_2 - \tau_1)$
2,3,5	30,60,90,120,150,180,210
7	30,60,90,120,150,180
10,12	30,60,90
15	30,60

Tab. 4.3: Comparación de distintas combinaciones de parámetros para el integrado y secuencial sobre las instancias de MITRE donde integrado y secuencial con cualquier combinación de parámetros encontraron solución con prueba de optimalidad. En total son 35 instancias sumariadas por fila.

Analizando la tabla se observa que en casos de optimalidad, el integrado en promedio mejora el porcentaje de cancelaciones, salvo con parámetros de CMB, que

analizaremos con más detalle en la Tabla 4.4. Esto tiene sentido dado que los pesos de la función objetivo buscaban reducir cancelaciones y el secuencial busca el óptimo en un espacio de soluciones reducido en comparación con el integrado, entonces si ambos llegan al óptimo con los mismos parámetros, el integrado va a obtener un valor objetivo igual o menor que el secuencial. Sin embargo, observar que al incluir mejoras de factibilidad, la diferencia de cancelaciones entre integrado y secuencial se reduce, pasando de casi un 2.5 % de diferencia en promedio con parámetros BASE a menos de 0.1 % con parámetros TAXI+HE+CMB. Es decir que al flexibilizar horas extra, viajes en taxi y cancelación de almuerzos, el secuencial logra acercarse al mismo porcentaje de cancelaciones que el secuencial. También notar que los mínimos y máximos coinciden para parámetros que incluyen HE y CMB.

En cuanto a las demoras, observar primero que el secuencial siempre demora en promedio 70 minutos. Por un lado esto es así porque en todas las filas se consideran las mismas instancias y en cada fila varían los parámetros de reprogramación de tripulación, que no afectan a la primera fase del secuencial donde se asignan horarios. Por otro lado, la fase de reprogramación de tripulación no está cancelando trenes que tuvieran demoras, si no el promedio de demoras bajaría. En comparación con el integrado se observan menos demoras en general, ya que el integrado utiliza más demoras para reducir cancelaciones y mejorar las demás métricas.

Los porcentajes de reasignación de tareas son muy similares en todos los casos, siendo levemente menores en el caso del secuencial. Lo mismo sucede con los minutos de reposicionamiento viajando como pasajero en servicios de la red, todos están en el orden de 90 minutos en promedio y alrededor de 200 como máximo.

La mayor diferencia se observa en el uso de taxis para regresar a la base, horas extra y cancelación de almuerzos. En todos los casos, el promedio resultó mejor para el integrado. En el caso particular de los regresos a la base en taxi, observar que el p50 (la mediana) da cero en todos los casos (celdas vacías), es decir que la mitad de las instancias no utilizó viajes en taxi. Teniendo en cuenta también el máximo y el promedio, es claro que los retornos en taxi se usan en poca cantidad o no se usan para nada, al menos en estas instancias. En el caso de las horas extra, con parámetros HE el integrado usó la mitad en promedio y en p50, y en el máximo el integrado utilizó aproximadamente un 70 % menos que el secuencial. En cambio, con parámetros TAXI+HE+CMB, el uso de horas extra de integrado y secuencial es muy parecido. Por último en el caso de cancelación de almuerzos, con parámetros CMB el secuencial llegó a cancelar 3 o más almuerzos en la mitad de las instancias, mientras que el secuencial canceló 1 o menos en la mitad de las instancias. Ambos coinciden en el máximo de 4 almuerzos. Con parámetros TAXI+HE+CMB, el número se reduce pero sigue siendo más alto en el secuencial. Es importante destacar que cada almuerzo cancelado corresponde al recreo de 45 minutos de una tripulación, entonces hay una gran diferencia entre cancelar 1 o 2 almuerzos.

En términos generales, se observa como cada relajación es utilizada principalmente por la segunda fase del secuencial para mantener o aumentar lo menos posible el porcentaje de cancelaciones que obtuvo en la fase 1. El integrado también las utiliza pero en menor medida, ya que pondera el uso con cancelaciones y demoras.

En particular al utilizar las tres relajaciones en TAXI+HE+CMB, tanto el integrado como secuencial obtuvieron resultados similares, pero el integrado logró mejorar todas las principales métricas. Sólo vemos un caso llamativo del máximo de distancia recorrida en taxi que en el integrado fue 26. Este máximo se da en dos instancias de bloqueo de 2hs ( $d_e \in \{5, 7\}$ ). Comparando los resultados no agregados de esas dos instancias para secuencial e integrado, vemos que efectivamente el integrado utiliza más distancia en taxi, pero cancela un almuerzo menos, no usa horas extra (el secuencial sí lo hizo), utiliza la mitad de reposicionamientos como pasajero y reduce las demoras en un 15 %, mientras que el porcentaje de cancelaciones es el mismo. Es decir que a pesar de ser un valor más alto de distancia en taxi, las demás

métricas asociadas al uso de tripulación resultaron mejores. En la siguiente sección se podrán observar los valores concretos para el caso  $d_e = 7$ .

En cuanto a tiempos, nuevamente se observa que el integrado se acerca al límite en algunos casos, pero en general en la mitad de las instancias fueron suficientes 10 segundos para llegar al óptimo. El secuencial en cambio tuvo tiempos considerablemente inferiores, de no más de 15 segundos.

Como se observó en la Tabla 4.3, cuando utilizamos los parámetros CMB, el secuencial tuvo un menor porcentaje de cancelaciones en promedio. Esto es llamativo ya que los pesos de la función objetivo priorizan minimizar cancelaciones y el algoritmo secuencial recorre un subconjunto de soluciones factibles de las que recorre el integrado, entonces ante casos de optimalidad, sería esperable que el integrado siempre sea mejor. Sin embargo, en la Tabla 4.4 se observan las tres instancias que bajan el promedio de cancelaciones del secuencial en la tabla anterior.

		Integrado					
$d_e$	$(\tau_2 - \tau_1)$	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	CMB (cant)
5	120	17.72	271	80	45.63	139	1.0
	210	17.05	404	224	26.75	148	
7	120	17.72	271	80	45.63	139	

		Secuencial					
$d_e$	$(\tau_2 - \tau_1)$	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	CMB (cant)
5	120	15.04	230	112	51.46	111	3.0
	210	15.7	372	0	38.22	79	4.0
7	120	15.04	230	112	52.43	111	3.0

Tab. 4.4: Instancias de CMB donde tanto el integrado como el secuencial encuentran solución y prueba de optimalidad, pero el secuencial cancela menos.

En las instancias de 120 minutos de bloqueo, la diferencia en cancelaciones es de 41 minutos. Las demoras y el porcentaje de reasignación son mayores para el secuencial, lo cual es razonable, ya que opera más servicios. En cuanto a minutos de reposicionamiento, vemos que el secuencial utilizó alrededor de 30 minutos menos (un 20% menos), pero en cuanto a cancelación de almuerzos, el secuencial canceló 3 cuando el integrado ninguno. Son estos almuerzos los que afectan principalmente al porcentaje de cancelaciones. Cada uno de esos almuerzos dura 45 minutos, y a cada minuto cancelado le asignamos un costo de 500. Es decir que cancelar los tres almuerzos tiene un costo de 67500. Por otro lado, a cada minuto cancelado de tren le asignamos un costo de 1500, con lo cual los 41 minutos que se ganaron con el secuencial, tienen un costo de 61500. Es decir, cancelar los 3 almuerzos es más costoso que los 41 minutos de tren basándonos en los pesos que utilizamos en la función objetivo, con lo cual la solución que obtuvo el integrado es efectivamente mejor que la que obtuvo el secuencial en términos de la función objetivo, aunque tenga peores porcentajes de cancelación.

Se puede realizar un análisis análogo para la instancia de 210 minutos de bloqueo.

### Análisis con distintas duraciones de bloqueo en TAXI+HE+CMB

En la Tabla 4.5 analizamos las mismas métricas de calidad que la tabla anterior fijando los parámetros de ejecución en TAXI+HE+CMB y desagregando los resultados para mostrar todas las instancias con  $d_e = 7$ . Es decir que de las 35 instancias agregadas en la Tabla 4.3, aquí sólo estamos viendo 6. Además incluimos las instancias con  $d_e = 7$  en las cuales no se probó optimalidad (duración de bloqueo de 210

y 240 minutos), que en la Tabla 4.3 habíamos omitido.

Además incluimos dos columnas extra:

- Cancelaciones: Suma de minutos cancelados de todos los trenes. Este es el valor absoluto asociado a la columna % Cancelaciones.
- % GAP: Al igual que en la Tabla 4.2, se refiere al que informa CPLEX.

Las celdas vacías representan valores nulos.

Integrado										
$(\tau_2 - \tau_1)$	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
30	14.71	119	20	35.0	22					1.28
60	10.96	115	20	53.33	51		16			3.3
90	10.32	133	364	50.56	111		24	1		8.04
120	15.04	230	132	63.11	76	26	0	1		63.74
150	15.3	289	120	47.58	158			0		78.81
180	14.42	307	140	31.43	107			1		253.67
210	17.05	404	220	29.3	186			1	2.92	t.l.
240	16.17	422	258	37.36	200	8		1	2.3	t.l.

Secuencial										
$(\tau_2 - \tau_1)$	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
30	15.95	129	46	31.67	0					1.52
60	10.96	115	0	45.33	51		24			2.21
90	10.32	133	56	51.69	111		38	1		3.75
120	15.04	230	112	65.05	150	21	22	2		6.25
150	15.3	289	0	48.39	139			2		5.97
180	14.42	307	0	37.14	107			4		9.24
210	15.7	372	0	36.94	214			4		14.6
240	16.17	422	0	36.78	90	21		3		17.1

Tab. 4.5: Instancias con  $d_e = 7$  y parámetros: TAXI+HE+CMB. Comparación entre integrado y secuencial utilizando los parámetros más permisivos, para todas las instancias con demora máxima de 7 minutos.

En la tabla se observa que con los parámetros más permisivos, tanto el integrado como el secuencial obtienen porcentajes de cancelación similares. De hecho sólo se diferenciaron en bloqueos de 30 y 210 minutos, en los demás casos cancelaron la misma cantidad de minutos (no necesariamente los mismos trenes).

En particular en el bloqueo de 30, analizando la solución vimos que el secuencial en la fase 1 canceló 114 minutos con 46 minutos de demoras, pero los servicios cancelados eran distintos de los que canceló el integrado con duración de 119 minutos. En la segunda fase el secuencial se vio obligado a cancelar un servicio adicional para cumplir las restricciones de tripulación, llevando el total de cancelaciones a 129 minutos. Es interesante notar que en este caso al secuencial no le sirvió usar taxi, horas extra ni cancelar almuerzos para lograr la factibilidad y debió cancelar ese servicio adicional.

Si observamos el bloqueo de 210 minutos, el secuencial canceló 32 minutos menos (mitad de un servicio completo). A pesar de que el integrado no llegó a probar optimalidad y a pesar de haber cancelado más, la solución que obtuvo tiene un valor objetivo menor que el secuencial. Esto se debe a la gran diferencia en cancelación de almuerzos, donde el secuencial canceló 4 y el integrado sólo 1. Este caso es análogo a los que observamos en la Tabla 4.4.

En el resto de las instancias (donde el porcentaje de cancelaciones fue el mismo entre integrado y secuencial) vemos que en general el integrado utilizó más demoras en los eventos, pero realizó menos reasignaciones de tareas y utilizó menos reposicionamiento en taxi, horas extra y cancelaciones de almuerzos. Esto se debe a dos razones. Por un lado, a los pesos elegidos en la función objetivo. Por otro lado, el secuencial en la primera fase optimiza únicamente cancelaciones y demoras, y en la segunda fase los horarios están fijos (i.e. no agrega demoras) y por el resultado vemos que tampoco agregó cancelaciones, entonces encuentra una reasignación de tripulación factible empeorando las demás métricas. Es decir que el integrado logra realizar un mejor uso de los recursos de tripulación.

Un caso destacable es el bloqueo de 120 minutos, donde el integrado utilizó más distancia en taxi que el secuencial (recorriendo 5 estaciones de distancia adicionales), pero sin embargo no usó minutos extra y canceló un único almuerzo, comparado con los 22 minutos extra y los 2 almuerzos cancelados del secuencial. Esto se debe a los costos de la función objetivo, ya que cancelar un almuerzo es mucho más costoso que la distancia adicional recorrida en taxi. Observar también que esta instancia estaba presente en la Tabla 4.4, donde utilizamos sólo el parámetro de CMB y el secuencial cancelaba menos. Al incluir los demás parámetros, el integrado logró igualar el porcentaje de cancelaciones usando viajes en taxi y cancelando un único almuerzo.

### Análisis con distintos parámetros de ejecución

En las tablas presentadas en esta sección comparamos el uso de las distintas mejoras de factibilidad para ciertas instancias fijas elegidas convenientemente, variando los parámetros de ejecución de la Sección 4.4.1.

En las Tablas 4.6, 4.7 y 4.8 mostramos los resultados sobre bloqueos de 2hs para demoras máximas de 3, 7 y 15 minutos respectivamente (una instancia por tabla). Luego en la Tabla 4.9 realizamos el análisis para un bloqueo de 90 minutos y demora máxima de 15. Las métricas de calidad son las mismas que en la tabla anterior, y nuevamente las celdas vacías representan valores nulos.

En la Tabla 4.6 (instancia de bloqueo de 2hs y  $d_e = 3$ ) podemos observar que con una demora máxima tan chica, los porcentajes de cancelación son los mismos para varias combinaciones de parámetros. Esto puede deberse a que la mayor limitación del secuencial está asociada a que fija cancelaciones y demoras sin tener en cuenta restricciones de tripulación, pero en este caso, al tener tan poco margen para demorar eventos, el integrado termina cancelando lo mismo. Esto muestra que las demoras juegan un rol importante para poder bajar el porcentaje de cancelaciones en el integrado.

En particular con parámetros BASE, el integrado redujo las cancelaciones compensando con demoras, reasignación y reposicionamiento. Al incluir viajes en taxi, el integrado dejó de usar demoras, usó menos reposicionamiento pero utilizó más viaje en taxi para lograr reducir más el porcentaje de cancelaciones. El secuencial en cambio sufre el problema de haber elegido mal los servicios que canceló en la fase 1, ya que en todos los casos cancela 271 minutos pero en la segunda fase con BASE y TAXI se ve obligado a cancelar más. Con el resto de los parámetros ambos cancelan los mismos minutos pero no los mismos servicios, y eso lleva a que el integrado mejore en otras métricas, especialmente en cancelación de almuerzos.

En la Tabla 4.7 (instancia de bloqueo de 2hs y  $d_e = 7$ ) nuevamente se observan resultados similares en cuanto a cancelaciones, sin embargo el integrado realiza un mejor uso de viajes en taxi, horas extra y cancelación del almuerzos. El secuencial usa mucho más taxi y horas extra cuando sólo se permiten de forma individual en TAXI y HE, aunque el porcentaje de cancelaciones sea el mismo que el integrado. El caso de CMB es el que ya analizamos en la tabla 4.4. Al permitir las tres cosas a la vez, se observa como se balancean más, pero el integrado obtiene mejores métricas.

Integrado										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	19.82	303	30	54.37	111					3.93
TAXI	17.72	271	0	54.37	76	26				14.73
HE	17.72	271	0	63.11	171		38			22.06
CMB	17.72	271	30	56.31	111			1		3.58
TAXI+HE+CMB	17.72	271	0	52.43	149	13	8	0		17.83

Secuencial										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	21.91	335	0	38.84	51					5.62
TAXI	19.82	303	0	50.49	89	13				6.27
HE	17.72	271	0	56.31	171		52			5.53
CMB	17.72	271	0	43.69	130			2		4.6
TAXI+HE+CMB	17.72	271	0	53.4	111	0	8	1		6.97

Tab. 4.6: Instancia con  $d_e = 3$  y  $(\tau_2 - \tau_1) = 120$ .

Integrado										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	17.72	271	80	45.63	139					11.34
TAXI	17.14	262	172	56.31	84	8				27.4
HE	17.14	262	480	54.37	139		16			68.97
CMB	17.72	271	80	45.63	139			0		11.94
TAXI+HE+CMB	15.04	230	132	63.11	76	26	0	1		63.74

Secuencial										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	21.32	326	112	42.72	102					5.23
TAXI	17.14	262	112	64.08	127	26				4.37
HE	17.14	262	112	64.08	120		138			7.04
CMB	15.04	230	112	52.43	111			3		3.98
TAXI+HE+CMB	15.04	230	112	65.05	150	21	22	2		6.25

Tab. 4.7: Instancia con  $d_e = 7$  y  $(\tau_2 - \tau_1) = 120$ .

Integrado										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	14.06	215	804	46.6	111				14.43	t.l.
TAXI	17.14	262	918	62.14	149	29			42.74	t.l.
HE	15.89	243	1340	51.46	250		86		41.98	t.l.
CMB	9.88	151	644	52.43	111			2	2.3	t.l.
TAXI+HE+CMB	17.72	271	1487	47.57	141	0	0	0	40.52	t.l.

Secuencial										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	16.15	247	476	44.66	102					7.28
TAXI	16.15	247	476	57.28	103	8				10.24
HE	11.97	183	476	66.02	171		118			25.02
CMB	9.88	151	476	54.37	111			3		5.98
TAXI+HE+CMB	9.88	151	476	62.14	122	21	22	2		8.17

Tab. 4.8: Instancia con  $d_e = 15$  y  $(\tau_2 - \tau_1) = 120$ .

Integrado										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	15.28	197	116	47.19	111				9.85	t.l.
TAXI	13.5	174	1328	57.3	120	16				240.66
HE	12.8	165	518	48.31	111		16			144.13
CMB	10.32	133	96	51.69	111			2		17.23
TAXI+HE+CMB	10.32	133	476	53.93	111		16	1		59.8

Secuencial										
Parámetros	% Cancelaciones	Cancel. (min)	Demoras (min)	% Reasig.	Reposic. (min)	Taxi (dist)	HE (min)	CMB (cant)	% GAP	Tiempo (seg)
BASE	20.25	261	56	31.46	79					3.69
TAXI	20.25	261	56	31.46	79	0				3.8
HE	12.8	165	56	50.56	111		38			4.67
CMB	10.32	133	56	51.69	111			3		3.47
TAXI+HE+CMB	10.32	133	56	55.06	111		38	1		4.38

Tab. 4.9: Instancia con  $d_e = 15$  y  $(\tau_2 - \tau_1) = 90$ .

En la Tabla 4.8 (instancia de bloqueo de 2hs y  $d_e = 15$ ) se observa principalmente que el límite de 5 minutos de ejecución empieza a resultar un problema para el integrado. En particular los 3 casos donde el GAP estuvo por encima de 40%, el porcentaje de cancelaciones terminó siendo peor que el secuencial, lo cual tiene sentido dado que posiblemente el integrado se encuentra lejos del óptimo. En el caso BASE el integrado disminuye las cancelaciones en un 2% aprox. llevando al doble

las demoras. Y finalmente en el caso CMB se obtiene el mismo porcentaje de cancelaciones, pero el integrado canceló un almuerzo menos. Una importante observación de este último caso es que de todos es el que más cerró el GAP, y si observamos en tablas anteriores también ocurre que con parámetros CMB el integrado resuelve el problema en menor tiempo. Sin embargo si analizamos la factibilidad en la Tabla 4.1, usando parámetros CMB resolvimos las mismas instancias que con BASE (todas salvo las instancias de bloqueos de 240 minutos). Entonces por un lado ganamos en tiempo y en reducción de cancelaciones, pero ante bloqueos largos podríamos no encontrar solución.

Por último en la Tabla 4.9 (instancia de bloqueo de 90m y  $d_e = 15$ ) vemos que al reducir el tiempo de bloqueo a 90 minutos, el integrado prueba optimalidad con casi todos los parámetros, salvo en BASE. Esto muestra como la duración del bloqueo afecta el tamaño del problema y por ende el tiempo que el integrado necesita para resolverlo. En cuanto a la calidad, con BASE, el integrado logró reducir 1 hora aproximadamente las cancelaciones (un tren completo) con respecto al secuencial. Esto posiblemente se debe a que el secuencial cancela en la primera fase ciertos servicios que luego no le sirven para asignar tripulación y se ve obligado a cancelar más servicios en la segunda fase. Con parámetros TAXI el integrado llega a reducir casi una hora y media de cancelaciones con respecto al secuencial, introduciendo una gran cantidad de demoras y utilizando viajes en taxi, mientras que el secuencial esencialmente obtuvo el mismo resultado que en BASE, probablemente porque no logró salvar ningún tren con los viajes en taxi. Con los demás parámetros se repite lo que vimos en tablas anteriores, los porcentajes de cancelación son los mismos pero el integrado usa menos horas extra y cancela menos almuerzos, haciendo un mejor uso del recurso de tripulación.

## 5. Conclusiones y trabajo futuro

### 5.1. Conclusiones

Basándonos en los resultados preliminares, observamos que hay muchos casos donde la ejecución del algoritmo integrado mejoró la calidad de la solución con respecto a una iteración del framework iterativo (algoritmo secuencial), siempre respetando el límite de 5 minutos, y no siempre probando optimalidad. Las mejoras de calidad pueden resultar valiosas dependiendo de la aplicación concreta, especialmente en cuanto a cancelación de servicios y de almuerzos de la tripulación.

El secuencial se destacó principalmente en el tiempo que le tomó encontrar una solución. La primera fase que sólo realiza reprogramación de horarios y reasignación de material rodante suele resolverse rápido, y al fijar las cancelaciones y horarios para la segunda fase, muchas variables de asignación de recursos quedan determinadas unívocamente y el problema de reasignación de tripulación se reduce considerablemente. Pero como vimos en varios casos, la selección inicial de servicios cancelados del secuencial puede no ser buena y en la segunda fase puede generar más cancelaciones de trenes o sumar muchos viajes en taxi, horas extra o cancelaciones de almuerzos.

Recordemos que el secuencial simula una primera iteración del framework iterativo pero agrega la posibilidad de cancelar trenes y modificar la asignación de recursos de cronograma y material rodante en la fase de tripulación. Con lo cual nos encontramos en un escenario más flexible que el framework iterativo original presentado en el Capítulo 1, por lo tanto suponemos que dicho framework debería arrojar peores resultados que los que se observan en nuestra implementación secuencial.

En las instancias que analizamos y con los límites de tiempo que fijamos, podríamos decir que en los casos de bloqueos de 3hs o menos, limitando demoras a 7 minutos o menos, el integrado es una buena opción, ya que prueba optimalidad en la mayoría de los casos y tiene buenos resultados en los demás. Incluso podríamos permitir hasta 10 minutos de demora, casos que no analizamos formalmente ya que arrojan resultados similares a los que mostramos de  $d_e = 7$ . En cambio para bloqueos largos, el secuencial resulta una mejor opción ya que por el límite de tiempo el integrado no llega a obtener mejores soluciones. Para casos de máxima demora de 12 y 15 minutos, en bloqueos de menos de 2hs el integrado arrojó buenos resultados ya que logra probar optimalidad en la mayoría de los casos, pero en 2hs o más el tiempo se vuelve una limitante fuerte y nuevamente tiene más sentido la opción secuencial.

Verificando nuestras hipótesis, al permitir mayor demora por evento y al aumentar la duración del bloqueo, el problema resulta más complejo de resolver, y como conclusión general podemos decir que a mayor complejidad conviene usar el secuencial para garantizar la solución en tiempo real y a menor complejidad conviene el integrado, que arroja mejores resultados, incluso cuando no prueba optimalidad.

En cuanto a los distintos parámetros de ejecución, sólo teniendo en cuenta la factibilidad, resulta muy importante permitir horas extra, ya que nos dió 100% de instancias resueltas tanto integrado como secuencial. Pero si también tenemos en cuenta la calidad del servicio y buscamos reducir al mínimo las cancelaciones, deberíamos utilizar todas las mejoras (taxi, horas extra y cancelaciones de almuerzos) ya que en general fue el que más redujo las cancelaciones.

## 5.2. Trabajo futuro

En esta sección contaremos brevemente algunas ideas pendientes de trabajo a futuro.

### 5.2.1. Experimentación

Como mencionamos en la Sección 4.1, el siguiente gran paso para poner a prueba nuestros algoritmos tanto integrado como secuencial es aplicarlo en la instancia de la red ferroviaria Holandesa utilizada en la literatura, por ejemplo en Veelenturf et al. [1]. Esta red es bastante más compleja que el ramal Mitre que utilizamos en este trabajo ya que no es lineal, tiene una gran cantidad de estaciones y conexiones entre las mismas. Además nos permitiría comparar nuestros resultados con la literatura.

Ya sea con la instancia del ramal Mitre o con la instancia Holandesa, una mejora a nuestra experimentación preliminar sería utilizar datos reales de asignaciones de tripulación. Por ejemplo en Dollevoet et al. Dollevoet et al. [3] se menciona que los horarios de las tripulaciones suelen tener ciertos períodos de recambio de turno donde un bloqueo produce más cancelaciones ya que no se dispone de una gran cantidad de personal. En nuestros experimentos no pudimos modelar este escenario ya que tuvimos que limitarnos a resolver las asignaciones que generáramos.

También queda pendiente la experimentación con distintos parámetros de CPLEX, para explorar distintas formas de explorar el árbol de B&C y otros algoritmos para la relajación inicial.

Por último, al integrar dos problemas con gran nivel de dificultad, la cantidad de parámetros a considerar crece significativamente y requieren un gran esfuerzo de experimentación. Por esta razón tuvimos que fijar unos cuantos parámetros en nuestros resultados preliminares. A futuro quisiéramos explorar por ejemplo cómo afecta a la calidad y el tiempo de ejecución la cantidad máxima de tripulaciones viajando como pasajeras en una tarea  $\mathbb{P}_{\text{pas}}$ , o los parámetros  $\tau_1^{mb}$  y  $\tau_2^{mb}$  que definen el rango horario del almuerzo.

### 5.2.2. Modelo matemático

En cuanto al modelo que utilizamos, hay algunas alternativas que quedan para probar a futuro.

En la Sección 3.7.1, vimos que se podría pensar en reposicionamiento en taxi entre cualquier par de tareas, y no solo para regresar a la base. Podríamos buscar alguna manera de incorporar esta idea al modelo sin agregar necesariamente todas las aristas, y ver si así obtenemos mejores soluciones. Por ejemplo podríamos incluir sólo los viajes en taxi a estaciones geográficamente cercanas.

Como mencionamos en el Capítulo 3, la jornada original de una tripulación puede ser activa (i.e. realizando tareas) o de reserva. En este último caso, la tripulación permanece en su estación base durante su horario laboral, disponible para ser utilizada. Hasta ahora sólo penalizamos si se realizaba una tarea en la jornada original pero no en la alternativa, para minimizar cambios en la jornada. Sin embargo esta penalización no aplica al uso de tripulaciones de reserva. El modelo actual no permite penalizar el uso de tripulaciones de reserva innecesarias fácilmente, entonces podríamos adaptarlo para poder penalizarlas.

Por otro lado, en los experimentos observamos una gran mejora en factibilidad cuando dejamos de forzar que cada tripulación retome su asignación inicial después de  $\tau_3$  y permitimos intercambiar las compleciones de jornada. Este es un frente interesante que se podría explotar aún más, por ejemplo desagrupando las tareas incluídas en cada  $\varphi_c$  utilizando distintas estrategias y reasignarlas también. Es decir

que resolveríamos un problema de asignación de tripulación sobre esas tareas que por ser no cancelables ni demorables tenemos fijados horarios y asignación de material rodante. Esto podría resultar en una mejora de factibilidad incluso sin usar horas extra, viajes en taxi o cancelación de almuerzos, tanto para el integrado como para el secuencial.

Por último, en este trabajo vimos estrategias para recortar temporalmente a la red y de esta manera reducir la cantidad de variables y restricciones consideradas por el modelo. En una red lineal y chica como la del Mitre es posible tener en cuenta la red completa en el modelo, pero en redes más complejas, puede ser necesario recortar espacialmente la red para resolver el problema integrado en un entorno del bloqueo. Esto implica definir condiciones de borde que garanticen factibilidad por fuera de la red incluso durante el bloqueo. Este enfoque agrega bastante complejidad al modelado y al preprocesamiento, pero debería reducir considerablemente las variables y restricciones en redes ferroviarias geográficamente grandes.

### 5.2.3. Algoritmo

En este trabajo utilizamos CPLEX para resolver el problema con un algoritmo Branch & Cut, pero desde ya que tiene sentido explorar metaheurísticas y/o utilizar el modelo propuesto como herramienta auxiliar para resolver parte del problema.

# Bibliografía

- [1] Lucas Veelenturf, Martin Kidd, Valentina Cacchiani, Leo Kroon, Paolo Toth. *A Railway Timetable Rescheduling Approach for Handling Large-Scale Disruptions*. In *Transportation Science*, Pages 1-22, 2015.
- [2] Agustín Mosteiro. *Programación Lineal Entera aplicada al problema de Reprogramación del Cronograma Ferroviario ante grandes interrupciones*. Facultad de Cs. Exactas y Naturales, Universidad de Buenos Aires, Buenos Aires, 2017.
- [3] Twan Dollevoet, Dennis Huisman, Leo Kroon, Lucas Veelenturf, Joris Wagenaar. *Application of an iterative framework for real-time railway rescheduling*. In *Computers & Operations Research* 78, Pages 203-217, 2017.
- [4] Daniel Potthoff, Dennis Huisman. *Column Generation with Dynamic Duty Selection for Railway Crew Rescheduling*. In *Transportation Science* Vol. 44, No. 4, Pages 493-505, 2010.
- [5] Daniel Potthoff *Railway Crew Rescheduling: Novel approaches and extensions*, Chapters 3, 4. PhD Thesis, Erasmus University Rotterdam, 2010.
- [6] Lucas Veelenturf, Daniel Potthoff, Dennis Huisman, Leo Kroon. *Railway crew rescheduling with retiming*. In *Transportation Research Part C* 20, Pages 95-110, 2012.
- [7] Dennis Huisman. *A column generation approach for the rail crew re-scheduling problem*. In *European Journal of Operational Research* 180, Pages 163–173, 2007.
- [8] Nikola Besinovic, Valentina Cacchiani, Twan Dollevoet, Rob M.P. Goverde, Dennis Huisman, Martin P. Kidd, Leo G. Kroon, Egidio Quaglietta, Joaquin Rodriguez, Paolo Toth, Lucas Veelenturf, Joris Wagenaar. *Integrated Decision Support Tools for Disruption Management*. In 6th International Conference on Railway Operations Modelling and Analysis - RailTokyo2015, No. 075, Pages 1-13, 2015.
- [9] Valentina Cacchiani, Dennis Huisman, Martin Kidd, Leo Kroon, Paolo Toth, Lucas Veelenturf, Joris Wagenaar. *An overview of recovery models and algorithms for real-time railway rescheduling*. In *Transportation Research Part B* 63, Pages 15-37, 2014.
- [10] Michael Schachtebeck, Anita Schöbel. *To Wait or Not to Wait—And Who Goes First? Delay Management with Priority Decisions*. In *Transportation Science* Vol. 44 No. 3, Pages 307-321, 2010.
- [11] Cameron Walker, Jody Snowdon, David Ryan. *Simultaneous disruption recovery of a train timetable and crew roster in real time*. In *Computers & Operations Research* 32, Pages 2077-2094, 2005.