

1

# The Hypergraph Network Simplex Algorithm & Railway Optimization

Ralf Borndörfer

joint work with Isabel Beckenbach and Markus Reuther

**4th ISM-ZIB-IMI MODAL Workshop on Mathematical Optimization and Data Analysis** Tokyo, ISM, 27.03.2019

#### Definition

A directed hypergraph is a pair H = (V, A) where V is a finite set of vertices and A is a family of hyperarcs.

A hyperarc  $a \in \mathcal{A}$  is a pair a = (t, h) of disjoint sets  $t, h \subseteq V$  sets of vertices, at least one of them non-empty;  $t \subseteq V$  is called the *tail* of a,  $h \subseteq V$  is the head.







#### Definition

Let D = (V, A) be a simple directed draph. A directed hypergraph based on D is a pair H = (V, A) where  $A \subseteq 2^A$  is a set of non-empty subsets  $a \subseteq A$  of vertex-disjoint arcs. A directed hypergraph based on some graph D is called graph-based.

#### Remark

A graph-based directed hypergraph is a directed hypergraph: For  $a \in \mathcal{A}$  let  $t(a) \coloneqq \{v \in V : \exists (v, w) \in a\}$  and  $h(a) \coloneqq \{w \in V : \exists (v, w) \in a\}$ .





#### Definition

Let  $H = (V, \mathcal{A})$  be a directed hypergraph based on a directed graph  $D, c \in \mathbb{R}^{\mathcal{A}}$  a vector of costs, and  $b \in \mathbb{R}^{V}$  of demands s.t.  $b^{T}1 = 0$ .

The minimum cost hyperflow problem (MCH) is the linear program



A vector  $x \in \mathbb{R}^{\mathcal{A}}$  that is feasible for this LP is a hyperflow (in *H*) (actually a circulation).



# The Minimum Cost Hyperflow Problem

- In contrast to the graph case, there might not exist an integral min cost hyperfow, even if all data is integral (see example later).
- Finding a minimum cost integral hyperflow is NP-hard, even if the hyperarcs consist of at most two arcs.
- ▶ If the underlying digraph *D* is connected and  $A \subseteq A$ , i.e., all arcs of the underlying digraph are also hyperarcs, then



has a solution if and only if  $b^T 1 = 0$ .



#### **Earlier work**

R. Cambini, G. Gallo, and M. G. Scutellà: Flows on hypergraphs. Mathematical Programming 78.2, p. 195-217 (1997).

#### However

- ▶ We heavily use that we work on graph-based hypergraphs.
- The algorithm for our setting is simpler and closer to the original network simplex.

#### Reference

I. Beckenbach: Matchings and Flows in Hypergraphs. PhD thesis, Freie Universität Berlin (2019).



#### Assumption

The underlying digraph *D* is connected and  $A \subseteq \mathcal{A}$ .

► Let  $M \in \{0, \pm 1\}^{V \times A}$  be the incidence matrix of H. The minimum cost hyperflow problem can then be written as

min  $c^T x$ , Mx = b,  $x \ge 0$ .

- If  $B = \{a_1, \dots, a_k\}$ , then let  $M_{\cdot B} \coloneqq (M_{\cdot a_1}, \dots, M_{\cdot a_k})$ .
- $\blacktriangleright rk(M) = |V| 1$
- B is a basis if and only if  $rk(M_{\cdot B}) = |B| = |V| 1$ .
- ▶ If B is a basis, then  $H[B \cap A] = (V, B \cap A)$  is a forest that contains  $|V| |B \cap A| = |V| (|B| |B \setminus A|) = |B \setminus A| + 1$  components.





- Let  $B \subseteq \mathcal{A}$  be s.t. |B| = |V| 1,  $B_1 \coloneqq B \cap A$ ,  $B_2 \coloneqq B \setminus A$ .
- $H[B_1]$  is a forest with  $|B_2| + 1$  components.
- For every tree of  $H[B_1]$ , choose a root r and denote its tree by  $T_r$ .
- Let R be the set of all such roots.





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

If B is a basis and  $r_1, r_2 \in R$  two roots, the system

$$M_{\cdot B}f = -e_{r_1} + e_{r_2}, x \ge 0$$

has a unique solution; we can send 1 unit of flow from  $r_1$  to  $r_2$ .







▶ If B is a basis and  $r_1, r_2 \in R$  two roots, the system

$$M_{\cdot B}f = -e_{r_1} + e_{r_2}, f \ge 0$$

has a unique solution; we can (in a unique way) send 1 unit of flow from  $r_1$  to  $r_2$  in  $H[B \cap A]$ .

- The unique flow of 1 unit from an arbitrary fixed root  $r^*$  to some other other root  $r \neq r^*$  is called *elementary*.
- We can send 1 unit of flow from  $r_1$  to  $r_2$  via an arbitrary intermediate root  $r^*$ , i.e., from  $r_1$  to  $r^*$  to  $r_2$ .
- We can also (easily) send 1 unit of flow inside of a tree.
- Any flow in  $H[B \cap A]$  is a superposition of elementary flows and flows on trees.



- Let B be a basis,  $T_r, r \in R$ , the rooted trees,  $r^* \in R$  a fixed root.
- For every  $r \in R \setminus \{r^*\}$  there is a unique hyperflow  $f_r$  in (V, B) that transports 1 unit from  $r^*$  to r.
- Let  $F \coloneqq (f_r)_{r \in R \setminus \{r^*\}} \in \mathbb{R}^{B \times R \setminus \{r^*\}}$  be the elementary hyperflow matrix whose r-th column contains this flow.
- ► F is easily reconstructed from  $F|B_2 = F_{B_2}$ . by recomputing the flows on the trees (but this takes time).





#### Example

$$r^* = r_1, \qquad f_{r_2} | B_2 = \begin{pmatrix} 1/2 \\ 1/4 \end{pmatrix}$$



#### Example

$$r^* = r_1, \qquad F_{B_2} = (f_{r_2}, f_{r_3})_{B_2} = \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix}$$





#### Example

$$r^* = r_1, \qquad E_{R \setminus \{r_1\}} = \begin{pmatrix} 1 & 2 \\ 1 & -2 \end{pmatrix}, \qquad F_{B_2} = \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix}$$



• Define  $E \in \mathbb{Z}^{R \times B_2}$  as

$$E_{ra} \coloneqq |V(T_r) \cap h(a)| - |V(T_r) \cap t(a)|.$$

#### Lemma

- $\blacktriangleright \quad E_{R \setminus \{r^*\}} = F_{B_2}^{-1}$
- ► B basis  $\Leftrightarrow$   $H[B_1]$  forest of  $|B_2| + 1$  components  $\land rk(F_{B_2}) = |B_2|$  $\Leftrightarrow$   $H[B_1]$  forest of  $|B_2| + 1$  components  $\land rk(E) = |B_2|$ .



#### Example







#### Example





### Elementary Hyperflow and Intersection Count Matrix

- ▶ Let B be a basis,  $T_r, r \in R$ , the rooted trees,  $r^* \in R$  a fixed root.
- For every  $r \in R \setminus \{r^*\}$  there is a unique hyperflow  $f_r$  in (V, B) that transports 1 unit from  $r^*$  to r.
- Let  $F \coloneqq (f_r)_{r \in R \setminus \{r^*\}} \in \mathbb{R}^{B \times R \setminus \{r^*\}}$  be the elementary hyperflow matrix whose r-th column contains this flow.
- F is easily reconstructed from  $F|B_2 = F_{B_2}$ . by recomputing the flows on the trees.
- Define an intersection count matrix  $E \in \mathbb{Z}^{R \times B_2}$  as

$$E_{ra} \coloneqq |V(T_r) \cap h(a)| - |V(T_r) \cap t(a)|.$$

18

 $\blacktriangleright \quad E_{R \setminus \{r^*\}} = F_{B_2}^{-1}$ 

# The Hyperflow Network Simplex Algorithm

**Input:** Hypergraph  $H = (V, \mathcal{A})$  based on digraph D = (V, A), cost  $c \ge 0$ , demand b s.t.  $b^T 1 = 0$ , feasible basic (hyper)flow x s.t.  $supp(x) \subseteq A$ , associated basis B and tree  $T_r$ .

**Output:** (Fractional) Minimum cost hyperflow x.

1. (BTRAN) Solve  $\pi^T M_{.B} = c_B^T$  and compute reduced costs

 $\bar{c}_a = c_a - \pi(V(h(a)) + \pi(V(t(a))).$ 

- 2. (PRICE) If  $\bar{c} \ge 0$  then output x, stop; else choose  $a^{in}$  s.t.  $\bar{c}_{a^{in}} < 0$ .
- 3. (FTRAN) Solve  $M_{\cdot B}f = -M_{a^{\text{in}}}$ .
- 4. (CHUZR) Choose  $a^{\text{out}} \in \operatorname{argmin}\left\{-\frac{x_a}{f_a}: f_a < 0, a \in B\right\}$ .
- 5. (UPDATE)

Go to 1.

6.

$$\begin{aligned} x_a \leftarrow \begin{cases} -x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a = a^{\text{in}} \\ x_a - x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a \in B \\ x_a, & a \notin B \cup \{a^{\text{in}}\} \end{cases} \text{, Update } B, R, \{T_r\}, M_B, F. \end{aligned}$$



# (FTRAN) $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$

- ▶ 1 unit of flow on entering arc  $a^{in}$  creates at tree  $T_r$  demand  $E_{ra^{in}} \coloneqq -(|V(T_r) \cap h(a)| - |V(T_r) \cap t(a)|)$
- Compute hyperflow on basic hyperarcs as superposition of elementary flows  $f|B_2 = F_{B_2} \cdot E_{R \setminus \{r^*\}a^{\text{in}}}$ .
- Compute associated flow on trees  $f|B_1$  and set  $f = \begin{pmatrix} f|B_1 \\ f|B_2 \end{pmatrix}$ .  $M_{\cdot B}f = -M_{\cdot a}in$  $a_1$  $a_2$  $r_3$



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

Example: (FTRAN)  $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$ 

- a<sup>in</sup> is the red arc
- $E_{Ra}^{in} = (1,0,-1)^T$  $r_2$  $a_1$ 1  $a_2$



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

 $r_3$ 

# Example: (FTRAN) $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$

- Compute demand on tree vertices induced by flow on basic hyerparcs.
- Total demands (including demands on entering hyperarc) sum up to 0 on every tree.



Example: (FTRAN) 
$$M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$$

► Compute flow on each tree by reverse BFS.



# The Hyperflow Network Simplex Algorithm

**Input:** Hypergraph  $H = (V, \mathcal{A})$  based on digraph D = (V, A), cost  $c \ge 0$ , demand b s.t.  $b^T 1 = 0$ , feasible basic (hyper)flow x s.t.  $supp(x) \subseteq A$ , associated basis B and tree  $T_r$ .

**Output:** (Fractional) Minimum cost hyperflow x.

1. (BTRAN) Solve  $\pi^T M_{.B} = c_B^T$  and compute reduced costs

 $\bar{c}_a = c_a - \pi(V(h(a)) + \pi(V(t(a))).$ 

- 2. (PRICE) If  $\bar{c} \ge 0$  then output x, stop; else choose  $a^{in}$  s.t.  $\bar{c}_{a^{in}} < 0$ .
- 3. (FTRAN) Solve  $M_{\cdot B}f = -M_{a^{\text{in}}}$ .
- 4. (CHUZR) Choose  $a^{\text{out}} \in \operatorname{argmin}\left\{-\frac{x_a}{f_a}: f_a < 0, a \in B\right\}$ .
- 5. (UPDATE)

Go to 1.

6.

$$x_a \leftarrow \begin{cases} -x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a = a^{\text{in}} \\ x_a - x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a \in B \\ x_a, & a \notin B \cup \{a^{\text{in}}\} \end{cases} \text{, update } B, R, \{T_r\}, M_B, F.$$



(BTRAN) 
$$\pi^T M_{\cdot B} = c_B^T$$

- Set  $\pi'_R$ := 0 and extend to potential  $\pi'$  s.t. all tree arcs have reduced cost zero.
- The basic hyperarcs have (preliminary) reduced costs

$$\bar{c}'_a = c_a - \pi'(V(h(a)) + \pi'(V(t(a)))_a)$$

Adjust potentials at trees (except  $r^*$ , i.e.,  $\pi_{r^*} = 0$ ) such that basic hyperarcs get zero reduced costs by setting

$$\pi_r \coloneqq \overline{c}_{B_2}^{\prime T} F_{B_2 r}$$
 and  $\pi_{V(T_r)} \leftarrow \pi_{V(T_r)}^{\prime} + \pi_r \cdot 1.$ 

• 
$$\bar{c}_a = \bar{c}'_a = 0, \quad a \in B_1$$
  
•  $\bar{c}_a = \bar{c}'_a - \sum_{r \in R \setminus \{r*\}} \pi_r(|V(T_r) \cap h(a)| - |V(T_r) \cap t(a)|)$   
 $= \bar{c}'_a - \sum_{r \in R \setminus \{r*\}} \bar{c}'^T_{B_2 r} E_{ar}$   
 $= \bar{c}'_a - \bar{c}'_a = 0, \quad a \in B_2$ 

 $\blacktriangleright \quad \pi^T M_{\cdot B} = c_{B_{\cdot}}^T$ 

• Set  $\pi'_R$ := 0 and extend to potentials  $\pi'$  such that all tree arcs have reduced cost zero.



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

The basic hyperarcs have preliminary reduced costs

$$\bar{c}'_{a_1} = 1 - (1 + 0) + (2 + 2) = 4$$
  
 $\bar{c}'_{a_2} = 1 - (1 + 1) + (1 - 1) = -1$ 





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

 Adjust potentials at tree roots (except r\*) such that basic hyperarcs get zero reduced costs by setting

$$\pi_{R\setminus\{r_1\}}^T := \bar{c}_{B_2}'^T F_{B_2} = (4,-1) \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix} = (1.75,2.25).$$

and raising all tree potentials according to the roots.



# The Hyperflow Network Simplex Algorithm

**Input:** Hypergraph  $H = (V, \mathcal{A})$  based on digraph D = (V, A), cost  $c \ge 0$ , demand b s.t.  $b^T 1 = 0$ , feasible basic (hyper)flow x s.t.  $supp(x) \subseteq A$ , associated basis B and tree  $T_r$ .

**Output:** (Fractional) Minimum cost hyperflow x.

1. (BTRAN) Solve  $\pi^T M_{\cdot B} = c_B^T$  and compute reduced costs

 $\bar{c}_a = c_a - \pi(V(h(a)) + \pi(V(t(a))).$ 

- 2. (PRICE) If  $\bar{c} \ge 0$  then output x, stop; else choose  $a^{\text{in}}$  s.t.  $\bar{c}_{a^{\text{in}}} < 0$ .
- 3. (FTRAN) Solve  $M_{\cdot B}f = -M_{a^{\text{in}}}$ .
- 4. (CHUZR) Choose  $a^{\text{out}} \in \operatorname{argmin}\left\{-\frac{x_a}{f_a}: f_a < 0, a \in B\right\}$ .
- 5. (UPDATE)

Go to 1.

6.

$$\begin{aligned} x_a \leftarrow \begin{cases} -x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a = a^{\text{in}} \\ x_a - x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a \in B \\ x_a, & a \notin B \cup \{a^{\text{in}}\} \end{cases} , \text{ Update } B, R, \{T_r\}, M_B, F. \end{aligned}$$

# **UPDATE:** Root Set & Elementary Hyperflow Matrix

- New basis  $B' \leftarrow B \cup \{a^{\text{in}}\} \setminus \{a^{\text{out}}\}, B'' \leftarrow B \cup \{a^{\text{in}}\}$
- Find new root set R' s.t.  $R' \ni r^*$  and  $|R\Delta R'| = 1$

• 
$$M_{\cdot B}f = -M_{a^{\text{in}}} \Leftrightarrow M_{\cdot B^{\prime\prime}} \begin{pmatrix} f \\ 1 \end{pmatrix} =: M_{\cdot B^{\prime\prime}}f^{\prime\prime} = 0$$

•  $f_r'' \leftarrow \begin{pmatrix} f_r \\ 0 \end{pmatrix}$ ,  $r \in R \cup R'$ , where  $M_B f_r = e_r - e_{r^*}$  if there is  $r \in R' \setminus R$ 

$$F'_{ar} \leftarrow f''_r(a) - \frac{f''_r(a^{\text{out}})}{f(a^{\text{out}})} f''(a), \quad a \in B'', r \in R'$$

 $\blacktriangleright \quad F'_{a^{\text{out}}r} = 0$ 

$$\blacktriangleright \quad M_{\cdot B'}F'_{B'r} = M_{\cdot B''}F'_{B''r}$$

$$= M_{\cdot B''} f_r'' - \frac{f_r''(a^{\text{out}})}{f(a^{\text{out}})} M_{\cdot B''} f''$$
$$= M_{\cdot B''} f_r''$$
$$= M_{\cdot B} f_r$$
$$= e_r - e_{r^*}$$

### Example: UPDATE

 $F = (-0.5, -0.5, 0.5, 0.5, 0.5)^T$ 





### Example: UPDATE

 $f = (0.5, -0.5, 0.5, 0.5, -0.5)^T$ 





# Example: UPDATE

►  $F = (-0.5, -0.5, 0.5, 0.5, 0.5)^T$ ,  $f = (0.5, -0.5, 0.5, 0.5, -0.5)^T$ 

• Ratio is  $-\frac{0.5}{-0.5} = 1 \Rightarrow F'_{a^{\text{in}}} = 1$ 

 $F'_{B \cap B'} = (-0.5, -0.5, 0.5, 0.5)^T - (-1) \cdot (0.5, -0.5, 0.5, 0.5)^T = (0, -1, 1, 1)^T$ 









33



# The Hyperflow Network Simplex Algorithm

**Input:** Hypergraph  $H = (V, \mathcal{A})$  based on digraph D = (V, A), cost  $c \ge 0$ , demand b s.t.  $b^T 1 = 0$ , feasible basic (hyper)flow x s.t.  $supp(x) \subseteq A$ , associated basis B and tree  $T_r$ .

**Output:** (Fractional) Minimum cost hyperflow x.

1. (BTRAN) Solve  $\pi^T M_{\cdot B} = c_B^T$  and compute reduced costs

 $\bar{c}_a = c_a - \pi(V(h(a)) + \pi(V(t(a))).$ 

- 2. (PRICE) If  $\bar{c} \ge 0$  then output x, stop; else choose  $a^{\text{in}}$  s.t.  $\bar{c}_{a^{\text{in}}} < 0$ .
- 3. (FTRAN) Solve  $M_{\cdot B}f = -M_{a^{\text{in}}}$ .
- 4. (CHUZR) Choose  $a^{\text{out}} \in \operatorname{argmin}\left\{-\frac{x_a}{f_a}: f_a < 0, a \in B\right\}$ .
- 5. (UPDATE)

Go to 1.

6.

$$\begin{aligned} x_a \leftarrow \begin{cases} -x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a = a^{\text{in}} \\ x_a - x_{a^{\text{out}}}/f_{a^{\text{out}}}, & a \in B \\ x_a, & a \notin B \cup \{a^{\text{in}}\} \end{cases} , \text{ Update } B, R, \{T_r\}, M_B, F. \end{aligned}$$

# Example: Start with a Feasible Hyperflow

Only arcs/hyperarcs of the current basis B are shown.

Flow 
$$x_a = \begin{cases} 1, & a \in B \\ 0, & a \notin B \end{cases}$$

• Costs c = 1, demands b as labeled.



• Set  $\pi'_R$ := 0 and extend to potentials  $\pi'$  such that all tree arcs have reduced cost zero.




### Example: (BTRAN) $\pi^T M_{\cdot B} = c_B^T$

The basic hyperarcs have preliminary reduced costs





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

## Example: (BTRAN) $\pi^T M_{\cdot B} = c_B^T$

 Adjust potentials at tree roots (except r\*) such that basic hyperarcs get zero reduced costs by setting

$$\pi_{R\setminus\{r_1\}}^T := \bar{c}_{B_2}^{\prime T} F_{B_2} = (4,-1) \begin{pmatrix} 1/2 & 1/2 \\ 1/4 & -1/4 \end{pmatrix} = (1.75,2.25).$$

and raising all tree potentials according to the roots.



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### Example: (PRICE) Choose $\bar{c}_{a^{in}} < 0$

- Green arc has reduced cost 1 4:25 + 2 = -1.25 < 0.
- Add this arc to the basis.





Example: (FTRAN)  $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$ 

a<sup>in</sup> is the red arc



#### Example: (FTRAN) $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$

- Compute demand on tree vertices induced by flow on basic hyerparcs.
- Total demands (including demands on entering hyperarc) sum up to 0 on every tree.



Example: (FTRAN) 
$$M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$$

► Compute flow on each tree by reverse BFS.





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

**Example:** (CHUZR)  $a^{\text{out}} \in \operatorname{argmin}\left\{-\frac{x_a}{f_a}: f_a < 0, a \in B\right\}$ 

•  $a^{\text{out}} = \text{red arc.}$ 







Example: (FTRAN)  $M_{\cdot B}f = -M_{\cdot a^{\text{in}}}$ 

a<sup>in</sup> is the red arc





#### **Example:** UPDATE

- $\blacktriangleright \quad x \leftarrow x + 1 \cdot f.$
- Trees  $T_{r_1}$  and  $T_{r_3}$  change.
- $R, B_2, F$  do not change.





- One iteration of the Hyperflow Network Simplex Algorithm can be implemented in  $O(\sum \deg(v) + |V|^2)$ .
- ► If Bland's rule is used, the algorithm is finite. Also polynomial?
- Generalization of Network Simplex Method to Graph Based Hypergraphs.
- Combinatorial: Each pivot consists of (fast) graph theoretical computations.
- Can handle upper bounds on the variables.





### InterCity Express (ICE) High Speed Train







#### Railway Constraints

Wagenstandanzeiger Gleis 11

Zeit	Zug		Richtung	G			D	[6]	E		4
00.34	EN	Jan Kospura	Roman GL Warszawa /		201 200 4	54 - CO - C				na di	
05.36	IC		Warechau Eleaunichweig Magheburg Lebbg / Hula Flugh	1							) <b>→</b>
06.21	ICE	Zugleburg in Harms	Leiping D bis 0 Köln / Bonn Flughaten A bis 0		The form the form	Anno Anno Anno Anno Anno Anno Anno Anno					
06.40	IC		Kolm Censtrikok Blad Benthelm Hongelu	Contraction of Care State				Line ST	the below	the besture	
07.45	IC 2234	Denstag bis Donverstag	Amaterdam Centraal								1
07.45	IC	Montag und Freitag	Distant.								
08.45	IC		Verden Bremen Deimenhonst					9 545 8 10		22	
09.40	IC		Dielefolg Dortmand Eisen							2	
10.45	IC	Dathiesland	Draman Oldenburg Dirochteich Mein			+ - LE 1				121 A	
11.40	1C		Elevented Gilterstoh Hanne			+ - 18 5		·	2* <b>1 1 1 1</b>	28	
12.45	IC 2130		Verden Bremen Deimenhon			+ (* A* A*	· M* · A	· P. · · P.	* <b>B</b> .*	12 C	
Re	XC	JUK	DHITV	/		+		- 22 - 2	2" EC"	e c	-
14.45	IC		Verden Dramen Deimenhonst Oldenburg			+ @	· M* • M			e c	-
15.31	ICE	Zugtallung in Hamm	O bis G Köln / Bonn Flughafen A bis C Köln	+							
16.45	IC		Bromen Okanburg Endan Norddeich Mole			+ - 1 1	· 🛛 * 🛛 🛤		2* ER2*	a -	3
17.40	IC		Doctrand Essen Duisburg Kóln			+ @			** [B]?*	3 6	3
18.45	IC		Verden Bremen Demenhoret Ordentizere			+ (* LT L		21 1	** [M] **	E -	3



Photos courtesy of DB Mobility Logistics AG

Train Composition

Maintenance

Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

 $\sim$ IO

### Timetable Regularity

Wagenstandanzeiger Gleis 11											
	-		9								
Zeit	Zug		Richtung	G	E	E	D	C	B		A
00.34	EN	Jan Kiepura	Rogen Poenan (8 Warszawa / Warszawa /	1	- 200 ·	24 · 20 · 2					-
05.36	IC		Braunechweig Mageicturg Leterg / Halla Flugh	1 1 1							]+
06.21	1CE	Zugleitung in Hamm	D bis 0 Köln / Bonn Flughaten A bis C		- 10,100 10,100	Hard Star Star	All of the sector				***
06.40	140		Koln Osražršick Blad Bertheim Hengelo	Comments of Lang				table of the later of the	Tele Ballantes	ballandare	
07.45	IC	Dienstag bie Domenstag			,						1
07.45	IC	Montag und Freitag	Bremen		÷(						_
08.45	IC		Bremen Bremen								-
09.40	IC		Oldenburg Dielefuld Dorthund			+ ~			R. MC.		
10.45	IC	Outhiesland	Esten Düsseldorf Dramen Oldenburg			+ <u> </u>					
11.40	IC IC		Eriden Norddaich Meia Bielefeld Gibersich			+ 2			2" AC		
12 45	2046		Hanne Dortmund Verden			+ C II = 6			2° # 2°	2	
13.40	2130		Deimenhorel Oldenburg Bielefeld			€ C 🔤 😫	* <u>• • • •</u>		an Mila	2 (	3
14.45	2040		Dortmund Esten Disseldorf Verden			+ C 11 2 1	" <u>R</u> " <u>R</u>		2" <b>P</b> C"	B (*	<u>ರ</u> ು
14.40	2038	Austalium in Harris	Bramen Detmenhorst Oldenburg		en bischen bischen	+ C	* . [A] * . [A]	2000	2. E.C.		3
15.31	ICE		Köln / Boss Flughafen A tris C Köln	+				<u></u>		* 周報	8. 8. 8. 2
16.45	1C 2004		Oldenburg Dinden Norddelch Mote			+ @ # .		8° 8	14 MIC 1	3 C	3
17.40	10		Essen Dulatsung Köln			+ @		P* 8	** <b>M</b> ?*	B C	3
18.45	5 IC		Verden Bremen Dememboret Oldenburg	famile !		• @	* 2 8		* <b>M</b> C*	E C	3
19.40	) IC	the Kiden	Rolefek) Götersich Hanm Dortmund	9.00.00		• C II I	* M* A		* 192*	27	3
20.45	5 IC		Varden Enemen Detrantorst	The second		• @ # 18	· A .		- MC	27 6	3

### Assignment Solution





#### Rotation Regularity





51

 $\bigcirc$ 

#### Modeling Regularity via Hyperedges





#### Hyperflow Model





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

A

#### Railway Constraints

Wagenstandanzeiger Gleis 11

Zeit	Zug		Richtung	G			D	[6]	E	-	-
00.34	EN	Jan Kingsons	Rospin Poenan GL Warszawa /		20 <b>3</b> 222 <b>4</b>	24 2 20 20 20				and and a	
05.36	IC		Beaurischweig Maghieurg Lebzig / Halle Flugh								<u>(``</u> )→
06.21	1CE	Zugleisung in Hamm	Clipping D bie O Köln / Bonn Flughaten A bis O		26 Julie 25 La	Hard Same Star Same Same					) →
06.40	140		Cenatricos Biad Bentheim Hengelo						inter beiten	the bardware	
07.45	IC 2234	Dienstag bis Domenstag	Amaterdam Central								I
07.45	IC	Montag und Freitag	Dreman							2	
08.45	IC EIM		Verden Bremen Delmenhorst							20	(
09.40	IC		Distributory Distributory Dottributory Ensert			+				29	
10.45	IC	Osthiesland	Braman Oldenburg Dirden			+ CLEAS					<u></u>
11.40	1C		Beiefeld Gütersich Hann Dortmund			+ (* 18 1.	- FA FA	·		12 9	-
12.45	IC BIN		Vierden Bremen Deimenhort Oldenburg			+ C B B		· P. * . P	12" MIC"	27	$ \Rightarrow $
Re	)(	JUIC	DHITV			+ @	M . H	- <b>P</b> - <b>P</b>		1	-
14.45	IC 2008		Verden Bremen Detmenhorst Oldentsorg			•	M .	2 PM 2 1 2 PM		1	
15.31	ICE	Zugtellung in Hamm	O bis G Köln / Bonn Flughaten A bis C Köln	+ 200						<b>1</b>	H H I
16.45	IC		Brensen Okamburg Einden Norddeich Mole			+ @ # #	· M* • M	28		2	
17.40	IC		Dudeturg Kóln			• @ # #			** 图(**	(B) (	=
18.45	1004		Barnen Detherhonti Oldestean			+ @ 11 11	AB		2" (U) 27	E (	



Photos courtesy of DB Mobility Logistics AG

Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

Aaintenance

Train Composition

# Rare Train Composition Example



#### Train Composition: Type, Order, Orientation



#### Hypergraph Model: Possible Train Compositions





#### Hypergraph Model: Arrival and Departure Nodes



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### Hypergraph Model: Single Traction



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### Hypergraph Model: Double Traction





#### Hypergraph Model: Triple Traction



Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### Hypergraph Model: Pass-Through Connections





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### Hypergraph Model: Pass-Through Connections





#### Hypergraph Model: All Connections





Hypernetwork Simplex Algorithm and Railways | 4th ISM-ZIB-IMI MODAL Workshop

#### The Coarse-to-Fine Method





### Motivation: ICE Connections



### Timetabled Trips: 1 Day



#### Timetabled Trips: Standard Week



#### Vehicle Rotation (1 Week)



#### Graphics: JavaView, MATHEON F4

69



#### Vehicle Rotation (6 Weeks)



#### Graphics: JavaView, MATHEON F4

70



#### Rotation Plan: Follow-on Trip Assignment

(Blue: Timetabled Trips, Red: Deadhead Trips)



#### Graphics: JavaView, MATHEON F4





#### Regular Timetable



Regularity of the timetable trips:

- Red: Trip done on one day of the week
- Blue: Trip done on all days of the week

#### Graphics: JavaView, MATHEON F4




## **Regular Rotations**



Regularity of the deadhead trips:

- ▷ Yellow: regular
- Red: irregular

retwo

Graphics: JavaView, MATHEON F4



## Thank you for your attention

Ralf Borndörfer

Zuse Institute Berlin Freie Universität Berlin Takustr. 7 14195 Berlin Germany

Fon (+49 30) 84185-243 Fax (+49 30) 84185-269

borndoerfer@zib.de

http://www.zib.de/borndoerfer



## THE POWER OF COOPERATION



