New Perspectives on PESP: T-Partitions and Separators

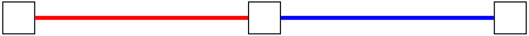
Niels Lindner — Zuse Institute Berlin
Christian Liebchen — TH Wildau

ATMOS 2019 @ Garching bei München
September 13, 2019

§1

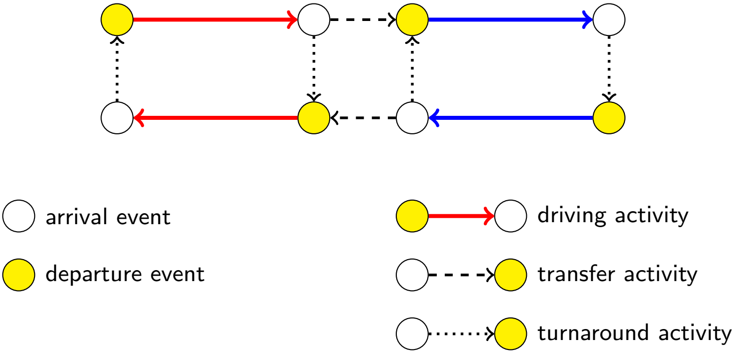
Introduction

Line Networks → Periodic Timetables



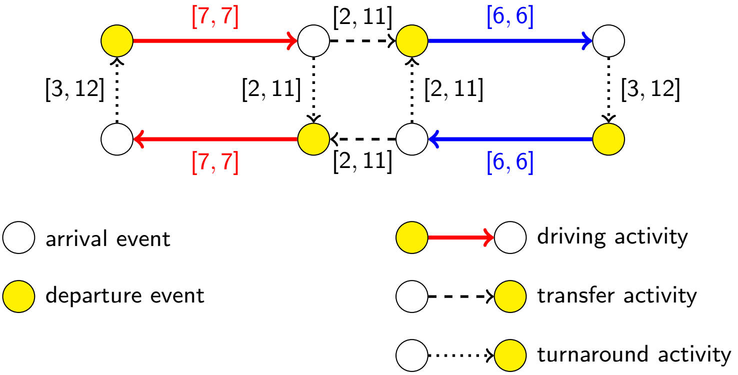
two lines meeting at a common station

Line Networks → Periodic Timetables



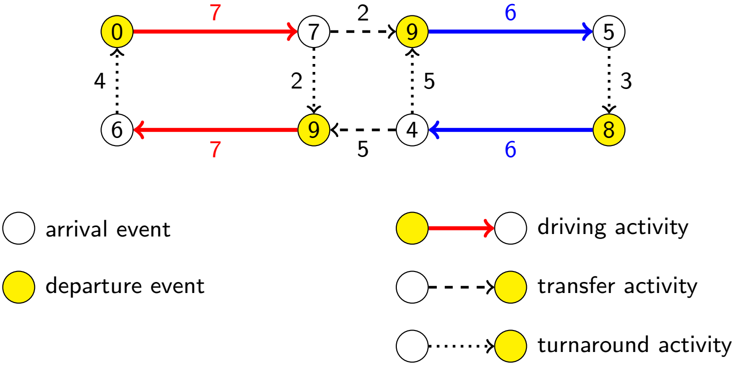
event-activity network model

Line Networks → Periodic Timetables



PESP instance (unweighted), period time $T = 10$

Line Networks → Periodic Timetables



periodic timetable, period time $T = 10$

Periodic Event Scheduling Problem

Serafini and Ukovich (1989)

Given

- a digraph $G = (V, A)$ (*event-activity network*),
- a period time $T \in \mathbb{N}$,
- lower and upper bounds $\ell, u \in \mathbb{Z}^A_{\geq 0}$, $\ell \leq u$,
- weights $w \in \mathbb{Z}^A_{\geq 0}$,

Periodic Event Scheduling Problem

Serafini and Ukovich (1989)

Given

- a digraph $G = (V, A)$ (*event-activity network*),
- a period time $T \in \mathbb{N}$,
- lower and upper bounds $\ell, u \in \mathbb{Z}^A_{\geq 0}$, $\ell \leq u$,
- weights $w \in \mathbb{Z}^A_{\geq 0}$,

the **Periodic Event Scheduling Problem (PESP)** is to find a *periodic timetable* $\pi \in \{0, 1, \dots, T-1\}^V$ and a *periodic tension* $x \in \mathbb{Z}^A$ such that

- $\pi_j - \pi_i \equiv x_{ij} \bmod T$ for all $ij \in A$,
- $\ell \leq x \leq u$,
- $w^t x$ is minimum.

Equivalently, minimize the weighted *periodic slack* $w^t y$, where $y := x - \ell$.

Solving PESP Instances

Solution Approaches

- branch-and-cut/mixed integer programming (Liebchen, Peeters, ...)
- modulo network simplex heuristic (Nachtigall, Opitz, Goerigk, ...)
- line cluster matching heuristic (Pätzold, Schöbel)
- Boolean satisfiability (Großmann, Nachtigall, ...)
- ...

Solving PESP Instances

Solution Approaches

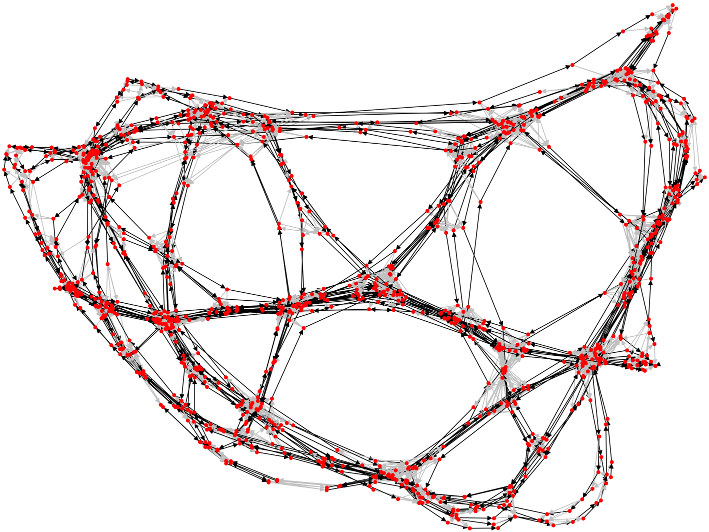
- branch-and-cut/mixed integer programming (Liebchen, Peeters, . . .)
- modulo network simplex heuristic (Nachtigall, Opitz, Goerigk, . . .)
- line cluster matching heuristic (Pätzold, Schöbel)
- Boolean satisfiability (Großmann, Nachtigall, . . .)
- . . .

PESPlib

- 20 very hard PESP instances
- no instance solved to optimality, current best gap: 34.64%
- maintained by Marc Goerigk

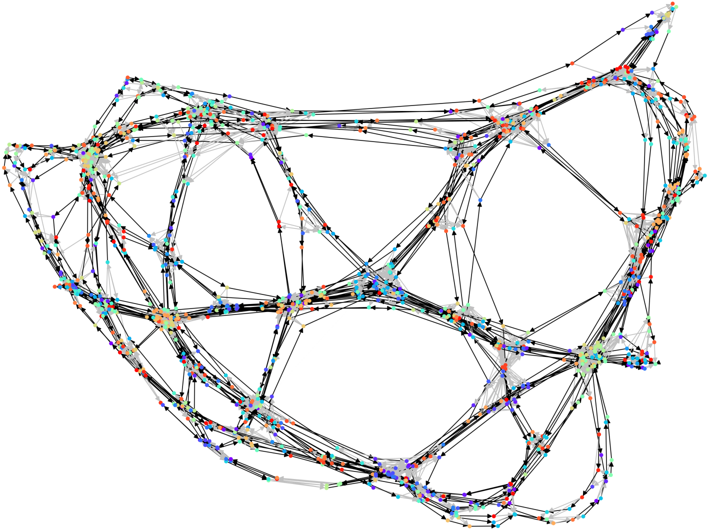
  `num.math.uni-goettingen.de/~m.goerigk/pesplib`

PESPlib: R1L1



after preprocessing: 1 214 vertices, 3 935 arcs, 2 722 linearly independent cycles

PESPlib: R1L1



current best timetable, weighted slack = 30 415 672

§2

T-Partitions

Trivial Observations

Timetables and T-Partitions

- Any vector $\pi \in \{0, \dots, T-1\}^V$ partitions the vertex set V into T possibly empty sets:

$$V = \dot{\bigcup_{d \in \{0,\dots,T-1\}}} \{v \in V \mid \pi_v = d\}.$$

- Conversely, any T-partition $(V_0, \dots, V_{T-1})$ of V yields a vector $\pi \in \{0, \dots, T-1\}^V$.

Trivial Observations

Timetables and T-Partitions

- Any vector $\pi \in \{0, \ldots, T-1\}^V$ partitions the vertex set V into T possibly empty sets:

$$V = \dot{\bigcup_{d \in \{0,\ldots,T-1\}}} \{v \in V \mid \pi_v = d\}.$$

- Conversely, any T-partition $(V_0, \ldots, V_{T-1})$ of V yields a vector $\pi \in \{0, \ldots, T-1\}^V$.

Addition and Subtraction modulo T

From any $\pi_1 \in \{0, \ldots, T-1\}^V$, we can obtain any $\pi \in \{0, \ldots, T-1\}^V$ by adding (mod T) some other vector $\pi_2 \in \{0, \ldots, T-1\}^V$.

Striving for Optimality

Lemma

For every $\pi \in \{0, \ldots, T-1\}^V$, there is a T-partition $\mathcal{V} := (V_0, \ldots, V_{T-1})$ such that the periodic timetable $\pi^{\mathcal{V}}$ defined by

$$\pi_v^{\mathcal{V}} := [\pi_v + d]_T, \quad v \in V_d, \quad d = 0, \ldots, T-1,$$

is feasible and has minimum weighted slack.

Striving for Optimality

Lemma

For every $\pi \in \{0, \dots, T-1\}^V$, there is a T-partition $\mathcal{V} := (V_0, \dots, V_{T-1})$ such that the periodic timetable $\pi^{\mathcal{V}}$ defined by

$$\pi_v^{\mathcal{V}} := [\pi_v + d]_T, \quad v \in V_d, \quad d = 0, \dots, T-1,$$

is feasible and has minimum weighted slack.

Maximally Improving T-Partition Problem

Find such a $\pi^{\mathcal{V}}$. I.e., for a given $\pi \in \{0, \dots, T-1\}$ with slack y, find a T-partition $\mathcal{V} = (V_0, \dots, V_{T-1})$ maximizing

$$\sum_{d=0}^{T-1} \sum_{e=0}^{T-1} \sum_{ij \in A \cap (V_d \times V_e)} w_{ij} \left(y_{ij} - [y_{ij} - d + e]_T \right)$$

subject to $[y_{ij} - d + e]_T \leq u_{ij} - \ell_{ij}$ for all $ij \in A$.

Delay Cuts

Remarks

- This relates PESP to Graph Partitioning and Minimum Cuts: As w.l.o.g. $y \leq T - 1$ and hence $y = [y]_T$, the contribution $y_{ij} - [y_{ij} - d + e]_T$ of arcs belonging to the same part of the T-partition to the objective function is 0.
- Finding a maximally improving T-partition is as difficult as solving PESP.
- Idea for heuristics: Restrict to special classes of T-partitions.

Delay Cuts

Remarks

- This relates PESP to Graph Partitioning and Minimum Cuts: As w.l.o.g. $y \leq T - 1$ and hence $y = [y]_T$, the contribution $y_{ij} - [y_{ij} - d + e]_T$ of arcs belonging to the same part of the T-partition to the objective function is 0.
- Finding a maximally improving T-partition is as difficult as solving PESP.
- Idea for heuristics: Restrict to special classes of T-partitions.

Delay Cuts (Goerigk, Schöbel: Multi-Node Cuts)

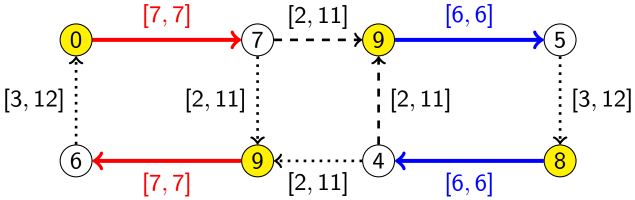
For $S \subseteq V$ and $d \in \{0, \dots, T-1\}$, the *delay cut* $\Delta(S, d)$ is defined as the T-partition

$$(V \setminus S, \emptyset, \dots, \emptyset, \underset{\substack{\uparrow \\ d}}{S}, \emptyset, \dots, \emptyset).$$

Intuitively, all events in S get delayed by d.

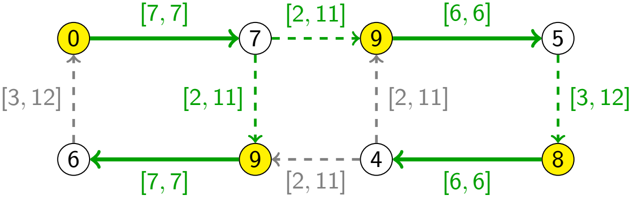
Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



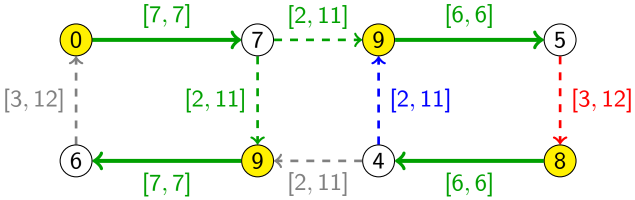
Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



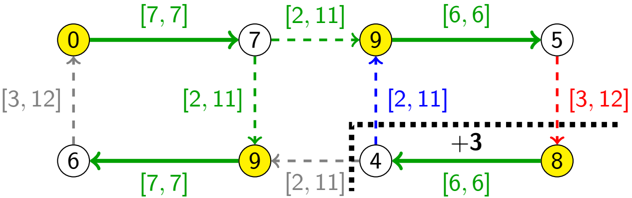
Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



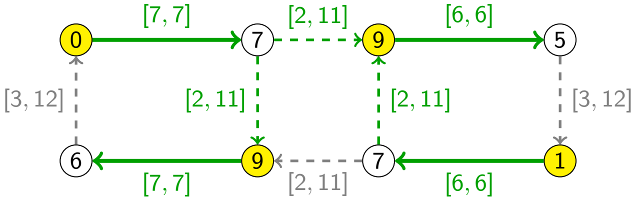
Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



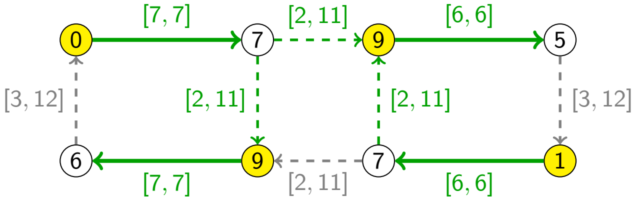
Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



Examples of Delay Cuts

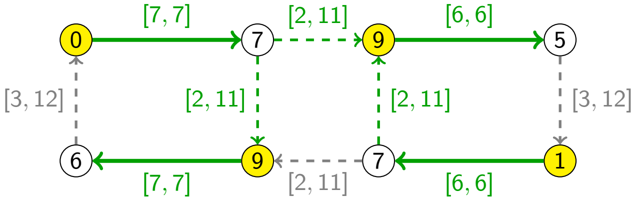
- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



- *Single-node cuts* (Nachtigall, Voget, 1996): Delay cuts with $|S| = 1$.

Examples of Delay Cuts

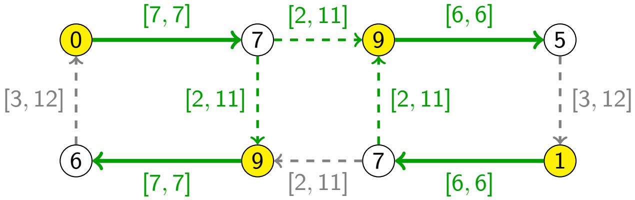
- *Modulo network simplex* (Nachtigall, Opitz, 2008):
  A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



- *Single-node cuts* (Nachtigall, Voget, 1996): Delay cuts with $|S| = 1$.
- *Waiting edge cuts* (Goerigk, Schöbel, 2012): Delay cuts with $|S| = 2$, the vertices of S are connected by an arc with small span $u - \ell$.

Examples of Delay Cuts

- *Modulo network simplex* (Nachtigall, Opitz, 2008): A move of the modulo network simplex is a delay cut corresponding to the fundamental cut of the forest arc. The delay depends on the co-forest arc.



- *Single-node cuts* (Nachtigall, Voget, 1996): Delay cuts with $|S| = 1$.
- *Waiting edge cuts* (Goerigk, Schöbel, 2012): Delay cuts with $|S| = 2$, the vertices of S are connected by an arc with small span $u - \ell$.
- *Multi-node cuts* (Goerigk, Schöbel, 2012): Delay cuts obtained by a greedy procedure.

Maximally Improving Delay Cuts

Maximally Improving Delay Cut Problem

For a given periodic timetable π, find the delay cut $\Delta(S, d)$ maximizing

$$\sum_{ij \in \delta^+(S)} w_{ij} \left(y_{ij} - [y_{ij} - d]_T\right) + \sum_{ij \in \delta^-(S)} w_{ij} \left(y_{ij} - [y_{ij} + d]_T\right)$$

such that $\pi^{\Delta(S,d)}$ is feasible.

Maximally Improving Delay Cuts

Maximally Improving Delay Cut Problem

For a given periodic timetable π, find the delay cut $\Delta(S, d)$ maximizing

$$\sum_{ij \in \delta^+(S)} w_{ij} \left(y_{ij} - [y_{ij} - d]_T \right) + \sum_{ij \in \delta^-(S)} w_{ij} \left(y_{ij} - [y_{ij} + d]_T \right)$$

such that $\pi^{\Delta(S,d)}$ is feasible.

Lemma

For fixed d, finding a maximally improving delay cut can be transformed to a standard maximum cut problem with linear objective, with both positive and negative weights.

Maximally Improving Delay Cuts

Maximally Improving Delay Cut Problem

For a given periodic timetable π, find the delay cut $\Delta(S, d)$ maximizing

$$\sum_{ij \in \delta^+(S)} w_{ij} \left(y_{ij} - [y_{ij} - d]_T \right) + \sum_{ij \in \delta^-(S)} w_{ij} \left(y_{ij} - [y_{ij} + d]_T \right)$$

such that $\pi^{\Delta(S,d)}$ is feasible.

Lemma

For fixed d, finding a maximally improving delay cut can be transformed to a standard maximum cut problem with linear objective, with both positive and negative weights.

Remarks

- Although NP-hard, the maximum cut problem can be solved within a reasonable amount of time (Borndörfer, Lindner, Roth, 2019).
- If there is no improving delay cut, then there is also no improving move for modulo network simplex, single- and multi-node cuts.

§3

Separators

Definition

Idea

- Divide and conquer!
- Top-down instead of bottom-up
  (Matching approach of Pätzold and Schöbel, ATMOS 2016)

Definition

Idea

- Divide and conquer!
- Top-down instead of bottom-up
  (Matching approach of Pätzold and Schöbel, ATMOS 2016)

Definition

Let (G, T, ℓ, u, w) be a PESP instance, $G = (V, A)$. For $\nu : 2^V \to \mathbb{R}_{\geq 0}$ and an imbalance $\alpha \geq 1$, a (ν, α)-*separator* is a subset $S \subseteq V$ s.t.

- $\delta(S)$ contains only free arcs, i.e., $ij \in A$ with $u_{ij} - \ell_{ij} \geq T - 1$,
- $w(\delta(S))$ is minimum,
- $\nu(V \setminus S) \leq \nu(S) \leq \alpha \cdot \nu(V \setminus S)$.

Definition

Idea

- Divide and conquer!
- Top-down instead of bottom-up
  (Matching approach of Pätzold and Schöbel, ATMOS 2016)

Definition

Let (G, T, ℓ, u, w) be a PESP instance, $G = (V, A)$. For $\nu : 2^V \to \mathbb{R}_{\geq 0}$ and an imbalance $\alpha \geq 1$, a (ν, α)-*separator* is a subset $S \subseteq V$ s.t.

- $\delta(S)$ contains only free arcs, i.e., $ij \in A$ with $u_{ij} - \ell_{ij} \geq T - 1$,
- $w(\delta(S))$ is minimum,
- $\nu(V \setminus S) \leq \nu(S) \leq \alpha \cdot \nu(V \setminus S)$.

Examples

- $\nu(X) := |X|$ (balances number of vertices)
- $\nu(X) := |A(G[X])| - |X| + 1$ ($\approx$ balances cyclomatic number)

Computational Aspects

Separating

- Computing optimal vertex-balanced cuts is NP-hard.
- Lots of heuristic software is available (e.g., Metis, KaHIP, FlowCutter), but usually only for simple ν.
- We provide mixed integer linear programs with $|V|$ binary variables for the vertex-balanced and cycle-balanced cases.
- Non-free arcs may be contracted, orientation of arcs is irrelevant.

Computational Aspects

Separating

- Computing optimal vertex-balanced cuts is NP-hard.
- Lots of heuristic software is available (e.g., Metis, KaHIP, FlowCutter), but usually only for simple ν.
- We provide mixed integer linear programs with $|V|$ binary variables for the vertex-balanced and cycle-balanced cases.
- Non-free arcs may be contracted, orientation of arcs is irrelevant.

Combining

- As all arcs in $\delta(S)$ are free in a (ν, α)-separator S, feasible timetables on the parts can always be combined to a feasible timetable on the original instance.
- We may apply a delay cut $\Delta(S, d)$ to S to get a better timetable, the slacks in the parts remain unchanged.
- If L_1, L_2 are lower bounds for the weighted slack on the parts, then $L_1 + L_2$ is a lower bound for the weighted slack on the whole instance.

Experiments

Instances

- 16 PESPlib railway instances RxLy
- 4 BLx bus instances omitted
  ($\leq$ 3 vertices left after contracting non-free arcs)

Experiments

Instances

- 16 PESPlib railway instances RxLy
- 4 BLx bus instances omitted
  ($\leq$ 3 vertices left after contracting non-free arcs)

Separators

- vertex- and cycle-balanced separators, $\alpha \in \{1.05, 1.1, 1.2, 1.5\}$
- initial solution by Metis (vertex case)
- MIP: Gurobi 8.1, 20 minutes, 8 threads

Experiments

Instances

- 16 PESPlib railway instances RxLy
- 4 BLx bus instances omitted
  ($\leq$ 3 vertices left after contracting non-free arcs)

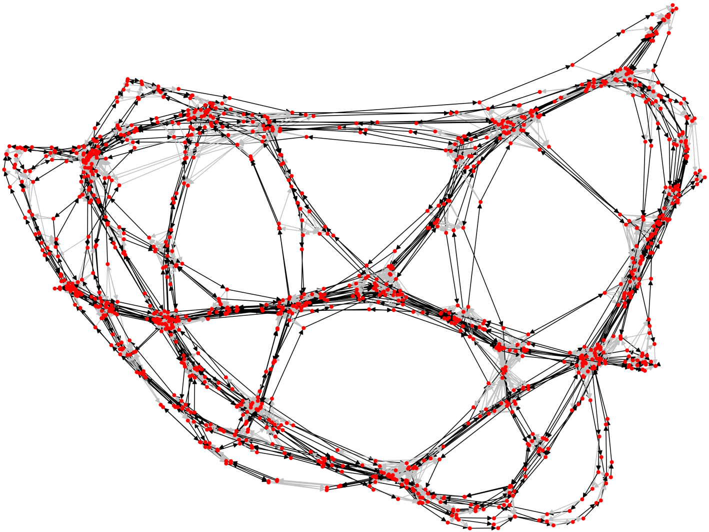
Separators

- vertex- and cycle-balanced separators, $\alpha \in \{1.05, 1.1, 1.2, 1.5\}$
- initial solution by Metis (vertex case)
- MIP: Gurobi 8.1, 20 minutes, 8 threads

Periodic Timetabling

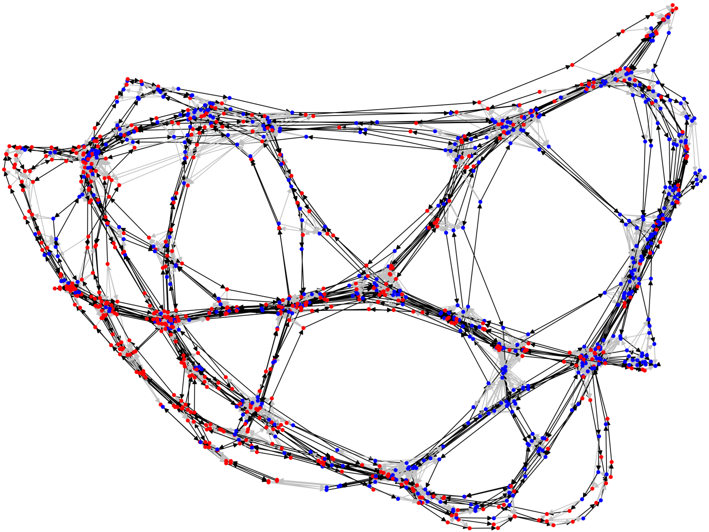
- concurrent PESP solver including MIP (CPLEX 12.8), MNS, Max Cut
- 10 minutes on each part vs. 20 minutes on full instance, 7 threads
- primal run and dual run

R1L1



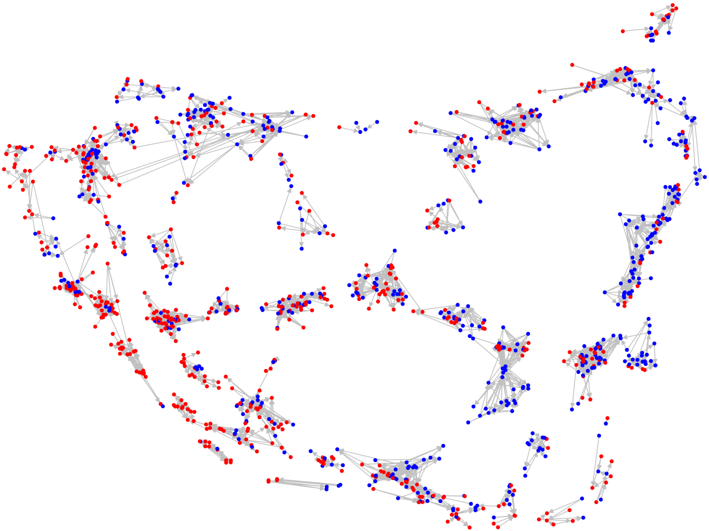
event-activity network after preprocessing, 2 722 linearly independent cycles

R1L1



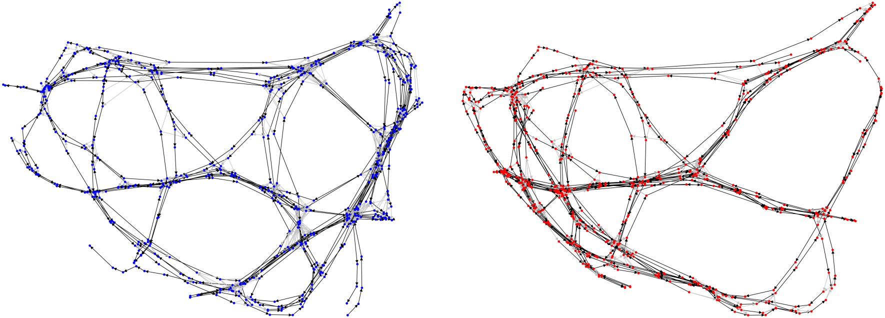
cycle-balanced separator, imbalance 1.2 (1.1975), weight 654 851, gap 36.3%

R1L1



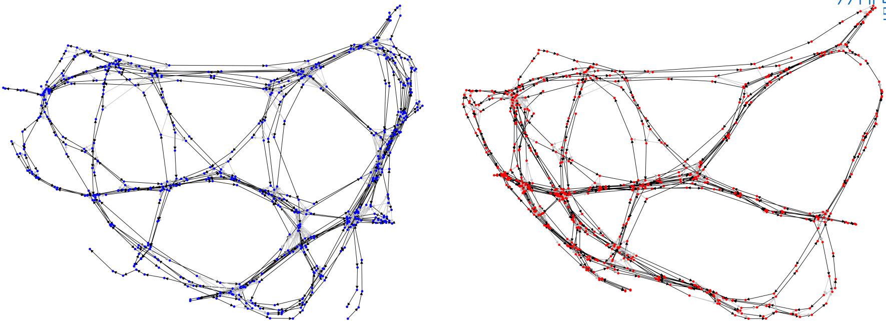
only free arcs

R1L1



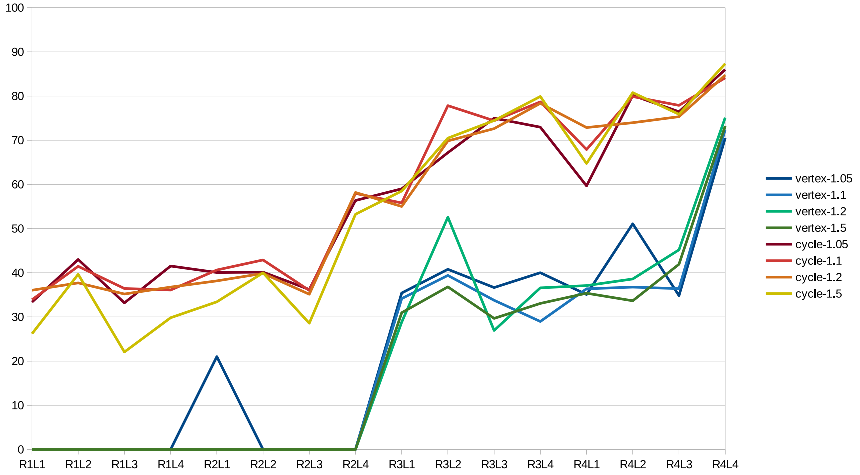
	left	right	cut	combined	original
cyclomatic number	782	653	466		2 722
weight	29 076 540	17 441 343	654 851		47 172 734
free weight	1 163 077	239 478	654 851		2 057 406

2 × 10 minutes of R1L1



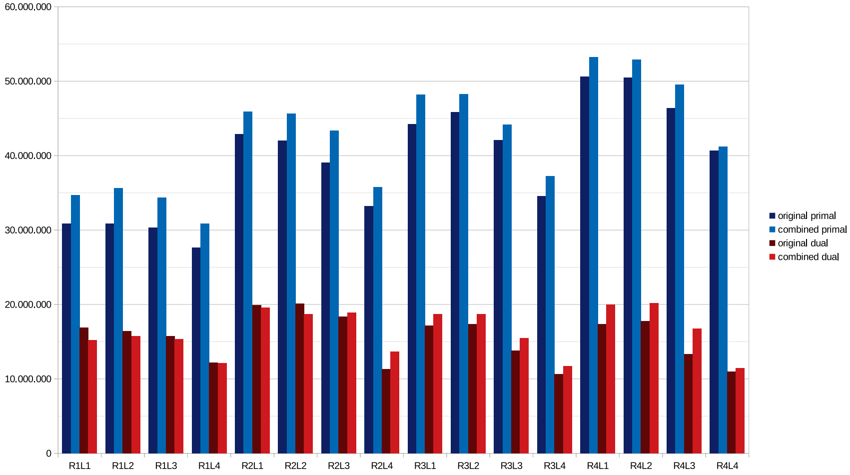
	left	right	cut	combined	original
cyclomatic number	782	653	466		2 722
weight	29 076 540	17 441 343	654 851		47 172 734
free weight	1 163 077	239 478	654 851		2 057 406
primal bound	15 029 848	2 985 689	16 653 876	34 669 413	30 861 021
dual bound	10 518 964	2 341 735	0	12 860 699	16 868 573
∅ free wt. slack	11.13	11.15	25.43	15.68	13.15
∅ non-free wt. slack	0.07	0.02	–	0.05	0.08

Optimality Gaps



optimality gaps for vertex- and cycle-balanced separators

Primal and Dual Bounds



primal and dual bounds, original (darker) and best combined (brighter)

The Last Slide

Conclusions

- The separator strategy has a structural disadvantage: The arcs in the cut receive a disproportionately high amount of slack.
- The cuts are too heavy in order to produce better primal bounds.
- However, the strategy pays off for dual bounds on larger instances.

The Last Slide

Conclusions

- The separator strategy has a structural disadvantage: The arcs in the cut receive a disproportionately high amount of slack.
- The cuts are too heavy in order to produce better primal bounds.
- However, the strategy pays off for dual bounds on larger instances.

Future Tasks

- find better functions ν, e.g., balancing the free weight
- close the optimality gap for (α, ν)-separators
- investigate methods for better dual bounds for PESP
- solve the parts to optimality
- more flexible combining of parts

New Perspectives on PESP: T-Partitions and Separators

Niels Lindner — Zuse Institute Berlin
Christian Liebchen — TH Wildau

ATMOS 2019 @ Garching bei München
September 13, 2019