none

Technische Universität Berlin

FAKULTÄT II
MATHEMATIK UND
NATURWISSENSCHAFTEN
Institut für Mathematik

On Cyclic Timetabling and Cycles in Graphs

by

Christian Liebchen Leon Peeters

No. 761/2002

ON CYCLIC TIMETABLING AND CYCLES IN GRAPHS

CHRISTIAN LIEBCHEN* AND LEON PEETERS†

Abstract. The cyclic railway timetabling problem (CRTP) essentially is defined by some constraint graph together with a cycle period time. We point out the relevance of cycles of the constraint graph for the CRTP. This covers valid inequalities for a Branch & Cut approach and special cases in that CRTP becomes easy. But emphasis will be put on the problem formulation.

The most intuitive model for cyclic timetabling involves node potentials. Modelling the cyclicity appropriately immediately results in an integer variable for every restriction, or arc of the constraint graph. A more sophisticated model regards the corresponding (periodic) tension. This well-established approach only requires an integer variable for every co-tree arc. The latter may be interpreted as representants for the elements of (strictly) fundamental cycle bases.

We introduce the more general concept of integral cycle bases for characterizing periodic tensions. Whereas the number of integer variables is still limited to the cyclomatic number of the constraint graph, there is a much wider choice for the cycle basis. One can profit immediately from this, because there are box constraints known for every integer variable that could ever appear.

Key words. periodic event scheduling problem, railway optimization, integral cycle basis, mixed integer programming

AMS subject classifications. 05C38, 90B06, 90C11

1. Introduction. In a cyclic timetable, connections are operated every cycle period at the same time instant. For example, the high speed ICE train from Cologne to Amsterdam reaches Amsterdam every two hours between 10:00 and 22:00 at 53 minutes past the hour.[1] Such cyclic timetables, also known as periodic timetables, are widely used in urban railway systems and in European national railway systems.

For the cyclic timetabling task, we are given a fixed set of lines in a traffic network. All the lines are operated with the same cycle period time. A cyclic timetable fixes the arrivals and departures of every line to instants within one abstract copy of the cycle period time. In order to get to a complete timetable, the abstract cycle period time has to be mapped to some clock time. In particular, the first and last trains of every line have to be fixed, and occasionally some additional rush-hour trips will be scheduled. In order to complete our sketch of the planning process of public transportation companies, we mention that the timetable serves as input for the vehicle scheduling, which usually is followed by the crew scheduling task. For a much more extensive description of that planning process, we refer to Bussieck et al. (1997).

Mathematical models and techniques for cyclic timetabling have been an actual topic for the last years. Because of the privatization of the European national and urban railways, railway operators are showing a growing interest in methods for improving and speeding up their timetable planning processes. Moreover, the increasing computational power of computers, together with the advances in mathematical models and solution methods, have made it technically possible to solve complex cyclic timetabling problems in a reasonable amount of time. Mathematical methods for constructing cyclic timetables are currently in use at NS Reizigers, The Netherlands (Hooghiemstra et al., 1999), and are being considered by the Berlin underground, Germany (Liebchen and Möhring, 2002).

*Technische Universität Berlin, Institut für Mathematik, Straße des 17. Juni 136, D-10623 Berlin, Germany. Email: liebchen@math.tu-berlin.de

†Universität Konstanz, D-78457 Konstanz, Germany. Email: lpeeters@fbk.eur.nl

[1] Actually, the train 16:56 is an exception and the 18:53 arrival is skipped.

Consequently, quite a bit of literature has been published in the area of cyclic timetabling the last decade. Most papers are based on the Periodic Event Scheduling Problem (PESP) introduced by Serafini and Ukovich (1989). Voorhoeve (1993) first used the PESP to model the problem of finding a feasible cyclic railway timetable. Schrijver and Steenbeek (1993) improved Voorhoeve's model and developed a constraint programming based algorithm. Nachtigall (1994, 1996a) first considered an objective function, namely minimizing the passenger waiting time. In a related paper (Nachtigall, 1996b), cyclic timetables consisting of train lines with different cycle times are studied. Nachtigall and Voget (1996) developed a genetic algorithm for minimizing the passenger waiting time. They also considered the bi-criterion objective of infrastructure investments versus passenger waiting time improvement (Nachtigall and Voget, 1997). Odijk (1996, 1997) studied the problem of composing several cyclic timetables for a station and its surroundings, so as to evaluate future infrastructure extension specifications.

An instance of a PESP based cyclic timetabling problem can be represented by a graph, the so-called constraint graph. This paper considers the relation between PESP based cyclic timetabling models and cycles in graphs. We first recall that periodic timetabling is easy as long as the constraint graph is a forest. Then we propose a transformation of the classical PESP based model to a formulation that is stated in terms of cycles in the constraint graph. The latter formulation contains an integer variable for every of the possibly exponentially many cycles in the constraint graph. We show when it suffices to consider only linearly many of these integer variables, and derive relevant properties of cycle bases. Further, some small examples illustrate the limits of our theory. Nachtigall (1996a) and Odijk (1996) showed that every cycle in the constraint graph induces several cutting planes for the PESP. Based on their findings, we present a class of cycles that induce strong cutting planes for the cyclic timetabling problem.

2. Cyclic Timetabling Models. This paper considers the widely used PESP (Serafini and Ukovich, 1989) for modeling cyclic timetables through *periodic time window constraints*. A periodic time window constraint periodically relates a pair of variables v_i and v_j, which represent the arrival or departure times of two train lines at some network node. More precisely, v_i and v_j represent the time instants at which two *events* i and j take place. In modeling cyclic timetables, an event is formed by a triplet (train, node, arrival) or (train, node, departure). For example, v_i may represent the arrival time of the mentioned ICE high speed line in Amsterdam. In that case, the event i consists of the triplet (ICE, Amsterdam, arrival).

A periodic time window constraint, then, involves a pair of events, (i, j), and has the general form

$$v_j - v_i + Tp_{ij} \in [\ell_{ij}, u_{ij}]. \tag{2.1}$$

For the moment ignoring the term Tp_{ij}, this constraint says that event j should take place between ℓ_{ij} and u_{ij} minutes later than event i does. The interval $[\ell_{ij}, u_{ij}]$ is the *time window* for the constraint. But with a cycle time $T = 60$, event j takes place 10 minutes after event i if $v_j = 5$ and $v_i = 55$, even though $v_j - v_i = -50$. Because of this cyclic nature of the timetable, the constraint (2.1) should be taken modulo T. This is where the term Tp_{ij} enters. The variable p_{ij} is required to be integer, and therefore the term Tp_{ij} allows for adding or subtracting an arbitrary integer multiple to or from the left hand side of the constraint. Thus, the term Tp_{ij} models the cyclic nature of the constraint. Since (2.1) is a periodic constraint, we assume $0 \le \ell_{ij} < T$, and $0 \leq u_{ij} - \ell_{ij} < T$.[2] The former assumption just scales the lower bound of the time window, and the latter assumption prevents meaningless time windows. Finally, we are interested in a timetable for one cycle period T, and therefore have the bounds $0 \leq v_i < T$ for the real valued variables.

2.1. Railway Timetabling Example Constraints. The following examples briefly illustrate the modeling power of periodic time window constraints of type (2.1) for cyclic railway timetables. We use d to indicate departure events, and a for arrival events. Moreover, for clarity, we use indices $1, 2, 3$, rather than ij, for the integer variables p_{ij} in the periodic time window constraints.

Suppose that a train t_1 takes 20 minutes to run from A to B. This gives the constraint

$$v_{t_1,A,d} - v_{t_1,B,a} + Tp_1 \in [20, 20].$$

Next, suppose that upon arrival in B, train t_1 has to stop between 1 and 3 minutes for the boarding and alighting of passengers before departing again. This gives

$$v_{t_1,B,d} - v_{t_1,B,a} + Tp_2 \in [1, 3].$$

Finally, suppose that, upon arrival in B, passengers need to be able to transfer from train t_1 to a connecting train t_2. Moreover, we want to give the passengers between 2 and 5 minutes to transfer to the connecting train t_2. We then have

$$v_{t_2,B,d} - v_{t_1,B,a} + Tp_3 \in [2, 5].$$

For much more extensive examples we refer to Krista (1996) and Peeters (2003).

2.2. Formulating the PESP. Let V be the set of events to be scheduled, representing train arrival and departure times, with $|V| = n$. Further, let A be the set of event pairs for which we have a timetable constraint, with $|A| = m$. The PESP model for cyclic timetabling is then defined as

$$\begin{array}{lll} \textbf{PESP:} & \text{Find a solution} & (v, p) \\ & \text{satisfying} & v_j - v_i + Tp_{ij} \in [\ell_{ij}, u_{ij}] \quad \text{for all } (i,j) \in A \\ & & v \in [0, T)^n \\ & & p \in \mathbb{Z}^m \end{array}$$

The $\mathcal{NP}$-completeness of the PESP for integer valued event times v_i and variable cycle time T has been shown by Serafini and Ukovich (1989) by a reduction from the Hamiltonian Cycle Problem. For a fixed cycle time $T \geq 3$, $\mathcal{NP}$-completeness was shown by a reduction from the Vertex Coloring Problem (Odijk (1997)). From this reduction and a result of Stockmeyer (1973), one can easily deduce that the PESP with cycle time $T = 3$ is $\mathcal{NP}$-complete even for planar graphs. Finally, after substituting a fixed vector p, the above mathematical program just formulates a Shortest Path problem with distance label variables v_i. Hence, for a fixed vector p, the PESP is polynomially solvable. This shows that the integer variables p_{ij} form the hard part of the PESP.

[2] In fact, after elimination of redundancies, we have $0 < u_{ij} - \ell_{ij} < T$.

2.3. The Constraint Graph. Because all constraints in the PESP are defined on pairs of events, a PESP problem instance can be described by a so-called *constraint graph* G. G is a directed graph. Its node set is the event set V, and the arc set is formed by the constraint set A. So for each event pair (i, j), for which a constraint is defined, we have an arc in G. Each arc, in turn, is described by its time window $[\ell_{ij}, u_{ij}]$. The only other parameter in the problem is the cycle time T. Therefore, a PESP instance is completely described by the constraint graph $G = (V, A, l, u)$, together with the cycle time T. It can easily be seen that even optimizing a linear objective over a PESP instance is trivial as long as the constraint graph is a forest: We just have to perform depth first search for every connected component. Hence, the cycles in G are the challenging part in periodic scheduling.

In the remainder of the paper, we use both the terms node and event, and event pair and arc. We use single indices $i, j, k, \ldots$ to indicate nodes in V. Arcs in A are denoted either by the double index (i, j), with the subscript ij, or by the single index a. We consider cycles and paths in the underlying undirected graph, i.e. they do not need to be directed in G. Therefore, we use the following notation. We arbitrarily choose a direction for each cycle $C \in G$. With respect to that direction, C consists of forward and backward arcs. The sets of forward and backward arcs are denoted by C^+ and C^-, respectively. For paths, a similar notation is used. A path from s to t is denoted by P_{st}. The path P_{st} is directed from s to t, and the sets of forward and backward arcs in P_{st} are denoted by P^+_{st} and P^-_{st} respectively.

2.4. The Cycle Periodicity Formulation. The constraint graph gives rise to an alternative formulation for the PESP. In order to derive this formulation, consider a directed graph $G = (V, A)$. A function $f : V \to \mathbb{R}$ is often referred to as a potential. So the event time instant variables $v_i,\ i \in V$ form a potential. A function $x : A \to \mathbb{R}$ is a *periodic tension with period* T if, for some potential v and some integer vector $p \in \mathbb{Z}^m$,

$$x_a = v_j - v_i + Tp_{ij} \text{ for all } a = (i, j) \in A.$$

So we can associate a periodic tension x with a solution (v, p) to a PESP instance. A cycle $C \in G$ is said to have the *cycle periodicity property* if it satisfies

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = Tq_C, \text{ for some integer variable } q_C. \tag{2.2}$$

The following theorem gives a necessary and sufficient condition for checking whether a function $x : A \to \mathbb{R}$ is a periodic tension with period T, without having to construct a solution to the PESP.

THEOREM 2.1 (Nachtigall, 1994). *Given a directed connected graph $G = (V, A)$ and a period T, a function $x : A \to \mathbb{R}$ is a periodic tension with period T if and only if each cycle $C \in G$ has the cycle periodicity property.*

Proof. Let $x : A \to \mathbb{R}$ be a periodic tension. Summing the values $x_a = v_j - v_i + Tp_{ij}$ along a cycle C, the variables v_i cancel out, giving

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = \sum_{a \in C^+} Tp_a - \sum_{a \in C^-} Tp_a.$$

The right hand side in this equality is clearly an integer multiple of T. In fact, we have

$$q_C = \sum_{a \in C^+} p_a - \sum_{a \in C^-} p_a. \tag{2.3}$$

Now suppose a vector $x \in \mathbb{R}^m$ satisfies

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = Tq_C \text{ for all cycles } C \in G.$$

We construct a corresponding solution (v, p) to the PESP. Choose an arbitrary spanning tree H of G, and set $p_a = 0$ for all $a \in H$. Next, choose some arbitrary node s, and set $v_s = 0$. For all other nodes $i \in N$, set

$$v_i = \sum_{a \in P_{si}^+} x_a - \sum_{a \in P_{si}^-} x_a,$$

where P_{si} is the path in H.

For a tree arc $(i,j) \in H$, the thus constructed function $v : V \to \mathbb{R}$ satisfies

$$v_j - v_i = \sum_{a \in P_{sj}^+} x_a - \sum_{a \in P_{sj}^-} x_a - \Big(\sum_{a \in P_{si}^+} x_a + \sum_{a \in P_{si}^-} x_a \Big) = x_{ij}.$$

So, by our choice for p_{ij}, we obtain $x_{ij} = v_j - v_i + Tp_{ij}$.

For a non-tree arc $(i,j) \notin H$, adding (i,j) to the tree creates a cycle C. If C contains s, we have

$$x_{ij} + \Big(\sum_{a \in P_{si}^+} x_a - \sum_{a \in P_{si}^-} x_a \Big) - \Big(\sum_{a \in P_{sj}^+} x_a - \sum_{a \in P_{sj}^-} x_a \Big) = Tq_C,$$

since C consists of $P_{si}, (i,j)$, and P_{sj}. If C does not contain s, then the common part of the paths P_{si} and P_{sj} cancels out in the above expression. We therefore have $x_{ij} + v_i - v_j = Tq_C$. So, setting $p_{ij} = q_C$ yields $x_{ij} = v_j - v_i + Tp_{ij}$.

It follows that x is a periodic tension. □

A similar and well known result for classical, a-periodic, tensions is obtained when q_C is set to zero beforehand in the cycle periodicity property (2.2).

In fact, the proof of theorem 2.1 indicates a recipe for constructing a PESP solution (v, p) from a periodic tension x. If the periodic tension x moreover satisfies the time windows, that is, if $\ell_a \le x_a \le u_a$ for all $a \in A$, then it can be used to obtain a solution for the PESP. Thus, a solution for the PESP can also be obtained by solving the Cycle Periodicity Formulation (CPF) below.

CPF: Find a solution (x, q)

satisfying

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = Tq_C \quad \text{for all } C \in G \tag{2.4a}$$

$$\ell_a \le x_a \le u_a \quad \text{for all } a \in A \tag{2.4b}$$

$$x \in \mathbb{R}^m \tag{2.4c}$$

$$q \in \mathbb{Z}^c \tag{2.4d}$$

Here, c equals the number of cycles in G. A drawback of the CPF is that G can contain exponentially many cycles, so c can be an exponentially large number. In particular, the complete graph on n nodes has $\sum_{k=3}^{n} \binom{n}{k} \frac{(k-1)!}{2}$ simple cycles. Moreover, parallel arcs are allowed in periodic scheduling, and are in fact quite common in modelling cyclic timetables. The CPF contains c so-called *cycle periodicity constraints* (2.4a), and moreover c integer variables q_C. The PESP formulation, on the other hand, has

m periodic time window constraints and m integer variables. However, in section 3.2 we show that, under certain conditions, it suffices to require the cycle periodicity constraints for all cycles in a basis of the cycle space of G. This cycle space has dimension $k = m - n + 1$, where k is the cyclomatic number of G.

2.5. Bounds on the Cycle Integer Variables. The following important result for the PESP by Odijk (1996) gives lower bounds and upper bounds on the integer variables q_C in the CPF.

THEOREM 2.2 (Odijk, 1996). *A PESP instance defined by the constraint graph $G = (V, A, l, u)$ and a period T is feasible if and only if there exists an integer vector $p \in \mathbb{Z}^m$ satisfying the* cycle inequalities

$$a_C \leq \sum_{a \in C^+} p_a - \sum_{a \in C^-} p_a \leq b_C \tag{2.5}$$

for all (simple) cycles $C \in G$, where a_C and b_C are defined by

$$a_C = \left\lceil \frac{1}{T} \left(\sum_{a \in C^+} \ell_a - \sum_{a \in C^-} u_a \right) \right\rceil, \qquad b_C = \left\lfloor \frac{1}{T} \left(\sum_{a \in C^+} u_a - \sum_{a \in C^-} \ell_a \right) \right\rfloor.$$

Note that the assumption $p \in \mathbb{Z}^m$ is important. Lindner (2000) provides an infeasible constraint graph, for which a fractional vector p fulfills any of the above cycle inequalities. If such an example did not exist, Theorem 2.2 would prove PESP to reside in $\mathcal{NPC} \cap \text{co-}\mathcal{NP}$, which is empty unless $\mathcal{P} = \mathcal{NP}$.

From equation (2.3), one sees that the parameters a_C and b_C can be also used as lower bounds and upper bounds on the variables q_C in the CPF.

LEMMA 2.3. *The cycle periodicity variables q_C in the CPF are bounded by $a_C \leq q_C \leq b_C$.*

Proof. Consider the cycle periodicity constraints (2.4a), and use the time windows $\ell_a \leq x_a \leq u_a$ to obtain bounds on q_C. Because of the integrality of q_C, these bounds can be divided by T and rounded, giving a_C and b_C. ∎

3. Cycle Bases and Cyclic Timetabling. This section describes the importance of cycle bases for cyclic timetabling. First, the concept and notation of cycle bases for undirected and directed graphs are briefly reviewed. Next, we present a theorem that shows when it suffices to enforce the cycle periodicity constraints (2.4a) only for the cycles in a cycle basis of G. The remainder of the section explores some other properties of cycle bases that are relevant for the CPF.

3.1. Cycle Bases of Graphs. We first briefly review the concept of a cycle basis of a graph. For an in-depth coverage of the subject, see Deo (1982) or Bollobás (1998). In an undirected graph $U = (V, E)$, a cycle C is encoded by a so-called *cycle vector* φ_C defined as

$$\varphi_{C,e} = \begin{cases} 1 & \text{if } e \in C, \\ 0 & \text{otherwise.} \end{cases}$$

Arithmetic for cycle vectors in undirected graphs is considered over the field GF(2). The *cycle space of an undirected graph U* is the space spanned by the $\{0, 1\}$ cycle vectors φ_C of cycles $C \in U$. Hereby we obtain exactly the vectors that have an even degree at any node. The cycle space of an undirected connected graph has the cyclomatic number $k = m - n + 1$ as its dimension. Hence, in general we have up to $2^k - 1$ simple cycles in U. A set of cycles is called a *cycle basis*, if it is a basis of the cycle space of U. For a cycle basis B with cycle vectors $\varphi_1, \ldots, \varphi_k$, the *cycle matrix* Γ'_B is the $k \times m$ matrix with the cycle vectors $\varphi_1, \ldots, \varphi_k$ as rows.

In a directed graph $G = (N, A)$, a cycle C is encoded by a $\{0, \pm 1\}$ cycle vector γ_C. Such a cycle is not required to be directed, so it may contain forward and backward arcs. Choosing an arbitrary direction for the cycle, a cycle vector γ_C for a directed graph is defined as

$$\gamma_{C,a} = \begin{cases} 1 & \text{if } a \text{ is a forward arc in } C, \\ -1 & \text{if } a \text{ is a backward arc in } C, \\ 0 & \text{if } a \notin C. \end{cases}$$

Contrary to the undirected case, arithmetic is performed over the field $\mathbb{Q}$ for cycles in directed graphs. The *cycle space of a directed graph* G is the space spanned by the $\{0, \pm 1\}$ cycle vectors γ_C of cycles $C \in G$, and the cycle space of a directed connected graph also has dimension $k = m - n + 1$. A set of cycles of G is called a *cycle basis*, if it is again a basis of the cycle space of G. The *cycle matrix* Γ_B corresponding to the set of vectors $\gamma_1, \ldots, \gamma_k$ of a cycle basis B, is the $k \times m$ matrix with $\gamma_1, \ldots, \gamma_k$ as rows.

3.2. Expressing Cycle Periodicity through Cycle Bases. For characterizing a classical, a-periodic, tension, it suffices to require

$$\sum_{a \in C^+} x_a - \sum_{a \in C^-} x_a = 0$$

only for the cycles of an arbitrary cycle basis of G, instead of for all cycles in G (see, for example Deo, 1982, Bollobás, 1998). This section presents a similar result for periodic tensions. However, for periodic tensions, we must ensure that the used cycle basis is integral.

DEFINITION 3.1 (Integral Cycle Basis). *A cycle basis B of G is an* integral cycle basis *if every non-basic cycle is an integer linear combination of the cycles in B.*

Let us explain integral cycle basis in more detail. Consider a cycle basis $B = \{C_1, \ldots, C_k\}$ with basic cycle vectors $\gamma_1, \ldots, \gamma_k$. For any cycle D, let $(\lambda_1^D, \ldots, \lambda_k^D)$ be the unique linear combination of basic cycles that yields D. That is,

$$\gamma_C = \sum_{i=1}^{k} \lambda_i^D \gamma_i. \tag{3.1}$$

Then B is an integral cycle basis if λ^D is an integral vector for every cycle D.

The following theorem shows that it suffices to require the cycle periodicity constraint for the cycles in an integral cycle basis of G only.

THEOREM 3.2. *If the cycle periodicity property holds for every cycle in an integral cycle basis B of G, then it holds for every cycle in C.*

Proof. Let $B = \{C_1, \ldots, C_k\}$ be an integral cycle basis of G, with cycle vectors $\gamma_1, \ldots, \gamma_k$. Suppose that the cycle periodicity property holds for every cycle in B:

$$\sum_{a \in C_i^+} x_a - \sum_{a \in C_i^-} x_a = T q_i, \text{ for all } i = 1, \ldots, k, \tag{3.2}$$

where q_i is the integer variable for cycle C_i. Consider a non-basic cycle D, and let $\lambda^D = (\lambda_1^D, \ldots, \lambda_k^D)$ be the linear combination of basic cycles that span D, so

$$\gamma_D = \sum_{i=1}^{k} \lambda_i^D \gamma_i. \tag{3.3}$$

For the directed sum of periodic tensions along D we have

$$\begin{aligned} \sum_{a \in D^+} x_a - \sum_{a \in D^-} x_a = \sum_{a \in A} \gamma_{D,a} x_a = \sum_{a \in A} x_a \sum_{i=1}^{k} \lambda_i^D \gamma_{i,a} = \sum_{i=1}^{k} \lambda_i^D \sum_{a \in A} \gamma_{i,a} x_a \\ = \sum_{i=1}^{k} \lambda_i^D T q_i = T \sum_{i=1}^{k} \lambda_i^D q_i. \end{aligned}$$

Since B is an integral cycle basis, we have $\lambda^D \in \mathbb{Z}^k$. Together with assumption (3.2), this means that $\sum_{i=1}^{k} \lambda_i^D q_i$ is integer. Therefore, the cycle periodicity property also holds for any cycle $D \notin B$. □

So, as long as we use an integral cycle basis, it suffices to require cycle periodicity only for the k basic cycles, in order to characterize periodic tensions. Moreover, the solution to the mathematical program (2.4) may violate the cycle periodicity property for non-basic cycles, if the cycle periodicity constraints are required only for the cycles in a non-integral cycle basis. This may result in a timetable that violates some of the imposed periodic constraints. Appendix A illustrates this problem through an example constraint graph. It follows that integral cycle bases form the minimal subset of cycles that suffices for formulating the CPF (2.4).

To check integrality of a cycles basis B, Cramer's rule immediately provides a sufficient criterion. Consider the cycle matrix $\Gamma_B^t = [\gamma_1, \ldots, \gamma_k]$. Removing the rows that correspond to the arcs of some spanning tree, we obtain a regular $k \times k$ submatrix A of Γ_B^t. If $|\det(A)| = 1$, then the unique solution to the system $Ax = b$ is integer for every all-integer right hand side b. In particular, every cycle D is an integer linear combination of the basic cycles $C_1, \ldots, C_k$ if $|\det(A)| = 1$. In fact, the above determinant condition is both, necessary and sufficient, as follows from an elementary result in the theory of integral lattices (Schrijver, 1998). The following theorem summarizes the above.

Theorem 3.3. *A cycle basis B is integral if and only if any regular $k \times k$ submatrix of its cycle matrix Γ_B has determinant ± 1. The latter is already true, if only one regular $k \times k$ submatrix of Γ_B has determinant ± 1.*

3.3. Fundamental Cycle Bases for Directed Graphs. In this section, we follow the notation of Whitney (1935) when he introduced the concept of matroids. First some properties of fundamental and strictly fundamental cycle bases for directed graphs are described. Next, we show that these cycle bases are integral.

Definition 3.4 (Strictly Fundamental Cycle Basis). *A set B of k cycles in a graph G is a* strictly fundamental cycle basis *if there exists a spanning tree H of G, such that the non-tree arcs of H generate B.*

Lemma 3.5 (Berge, 1962). *Any non-basic cycle D is a $\{0, \pm 1\}$ linear combination of the cycles in a strictly fundamental cycle basis B.*

We present a proof of Lemma 3.5 because the same ideas lead to a more general result.

Proof. Each cycle in a strictly fundamental cycle basis B contains an arc that appears exclusively in it, namely the generating non-tree arcs of the cycle. Placing these k unique arcs in the first columns, the cycle matrix Γ can be written in the form $\Gamma = [I|N]$, where I is the identity matrix. Let the unique linear combination of basic cycles that yields the non-basic cycle D be denoted by $\lambda^D = (\lambda_1^D, \ldots, \lambda_k^D)$. That is, $\gamma_D = \sum_{i=1}^k \lambda_i^D \gamma_i$, or, equivalently

$$\gamma_D = \Gamma^t \lambda^D = \left[\begin{array}{cc} I & N \end{array}\right]^t \lambda^D = \left[\begin{array}{c} I \\ N^t \end{array}\right] \lambda^D = \left[\begin{array}{c} \lambda^D \\ N^t \lambda^D \end{array}\right].$$

So, for the first k elements of γ_D, we have $[\gamma_{D,1}, \ldots, \gamma_{D,k}] = [\lambda_1^D, \ldots, \lambda_k^D]$. This uniquely determines λ^D. Since $\gamma_D \in \{0, \pm 1\}^m$, it follows that $\lambda^D \in \{0, \pm 1\}^k$. ☐

Actually, the proof indicates the following scheme for constructing the linear combination $\lambda^D = (\lambda_1^D, \ldots, \lambda_k^D)$ that gives the cycle D:

$$\lambda_i^D = \begin{cases} 1 & \text{if the generating chord of } \gamma_i \text{ is a forward arc in } D, \\ -1 & \text{if the generating chord of } \gamma_i \text{ is a backward arc in } D, \\ 0 & \text{otherwise.} \end{cases}$$

DEFINITION 3.6 (Fundamental Cycle Basis). *A set B of k cycles in a graph G is a* fundamental cycle basis *if there exists an ordering $C_1, \ldots, C_k$ of the cycles in B such that $C_i \setminus (C_{i-1} \cup \cdots \cup C_1) \neq \emptyset$ for all $i = 2, \ldots k$.*

So, a cycle basis is fundamental if, for some ordering of the cycles, each cycle contains at least one arc that is not part of its predecessors in that ordering. It is easy to see that a strictly fundamental cycle basis is also fundamental. Indeed, since each cycle in a strictly fundamental cycle basis has a unique arc, it holds that for *any ordering* of the cycles in a strictly fundamental basis, we have $C_i \setminus (C_{i-1} \cup \cdots \cup C_1) \neq \emptyset$.

LEMMA 3.7. *Any non-basic cycle C is an integer combination of the cycles in a fundamental cycle basis B.*

Proof. Let the basic cycle vectors $\gamma_1, \ldots, \gamma_k$ be ordered according to the fundamentality definition. We arrange the cycle matrix Γ as follows. Row i contains cycle vector γ_{k-i+1}, and the columns are arranged such that the unique arc of cycle C_i is placed in column $k - i + 1$. Since each cycle in the basis contains at least one arc that is not contained in its predecessors, there may be multiple candidates for being placed in the 'unique' column. In that case, we arbitrarily choose one. Arranging the cycle matrix in this way, it is written as $\Gamma = [U|N]$, where U is an upper triangular matrix with all ones on the diagonal. Next, let the unique linear combination of basic cycles that yields D be denoted by $\lambda^D = (\lambda_1^D, \ldots, \lambda_k^D)$. That is, $\gamma_D = \sum_{i=1}^k \lambda_i^D \gamma_i$, or, equivalently

$$\gamma_D = \Gamma^t \lambda^D = \left[\begin{array}{cc} U & N \end{array}\right]^t \lambda^D = \left[\begin{array}{c} U^t \\ N^t \end{array}\right] \lambda^D = \left[\begin{array}{c} U^t \lambda^D \\ N^t \lambda^D \end{array}\right].$$

Consider the first k elements of γ_D. These give $(\gamma_{D,1}, \ldots, \gamma_{D,k}) = U^t \lambda^D$. The matrix U^t is lower triangular with all ones on the diagonal, and all other entries are 0 or ± 1. Because of $\gamma_D \in \{0, \pm 1\}^m$, by performing Gaussian back substitution one can see immediately that the vector λ^D is all-integer. ☐

THEOREM 3.8. *A (strictly) fundamental cycle basis is integral.*

Proof. See Lemmata 3.5 and 3.7. ☐

Strictly fundamental cycle bases show that the CPF is polynomially solvable for a fixed vector q. The columns of the cycle matrix Γ for a strictly fundamental cycle basis can be ordered such that $\Gamma = [I|N]$, where N is a network matrix (Schrijver, 1998), so Γ is totally unimodular then. Consequently, for a fixed vector q and integer valued time windows $[\ell_a, u_a]$, the CPF defines an integral polyhedron, and thus can be solved by LP-techniques. Hence, we obtain $\mathcal{NP}$-completeness of the PESP even without requiring integrality of the potential resp. tension variables explicitly.

This shows that the integer variables q_C form the hard part of the CPF, similar to the variables p_{ij} forming the hard part of the PESP. Notice that the cycle matrix of a general fundamental cycle basis does not need to be totally unimodular, as is shown in Appendix D.

Earlier papers such as Nachtigall (1994) only proposed strictly fundamental cycle bases for formulating the CPF. Clearly, integral cycle bases provide a much wider choice of cycles bases.

3.4. Relation of Cycle Bases for Undirected and Directed Graphs. Consider a directed graph $G = (V, A)$, and let $U = (V, E)$ be the underlying undirected graph. The *projection* of a cycle $C \in G$ is the cycle $C' \in U$. So, for the cycle vectors γ and φ of C and C', it holds that $\varphi = |\gamma|$. The theorem below describes the relations between cycle bases for undirected and directed graphs.

THEOREM 3.9. *Consider a directed graph $G = (V, A)$ with underlying undirected graph $U = (V, E)$. Suppose that the set of undirected cycles $C'_1, \ldots, C'_k \in U$ are the projections of a set of cycles $C_1, \ldots, C_k \in G$. If $C'_1, \ldots, C'_k$ form a cycle basis of U, then $C_1, \ldots, C_k$ form a cycle basis of G.*

Proof. Let $\varphi_1, \ldots, \varphi_k$ be the cycle vectors of $C'_1, \ldots, C'_k$. Since $C'_1, \ldots, C'_k$ form a cycle basis of U, we have the following

$$\sum_{i=1}^{k} \lambda_i \varphi_i = 0 \text{ modulo } 2 \Leftrightarrow \lambda_i = 0 \text{ modulo } 2 \text{ for all } i = 1, \ldots, k. \tag{3.4}$$

Next, let $\gamma_1, \ldots, \gamma_k$ be the cycle vectors of $C_1, \ldots, C_k$. As remarked above, the cycle spaces of U and G both have dimension k. Therefore, if $\gamma_1, \ldots, \gamma_k$ do *not* form a basis of the cycle space of G, there must exist some $\lambda \neq 0$ such that

$$\sum_{i=1}^{k} \lambda_i \gamma_i = 0. \tag{3.5}$$

We assume that λ is integer, and that it contains at least one odd λ_i. Both assumptions are without loss of generality, since one can always construct an integer λ by multiplying by a sufficiently large number, and if all λ_i's are even, one can divide them by the smallest common power of 2.

Using the fact that γ_i modulo $2 = |\gamma_i| = \varphi_i$, expression (3.5) taken modulo 2 gives

$$\sum_{i=1}^{k} \lambda_i \varphi_i = 0 \text{ modulo } 2. \tag{3.6}$$

Since λ is integer, with at least one odd element, (3.6) contradicts (3.4). It follows that $C_1, \ldots, C_k$ must form a cycle basis of G. □

However, in general the reverse of Theorem 3.9 does not hold, as is shown by the example in Appendix B. But for integral cycle bases, we know that they project onto undirected cycle bases.

THEOREM 3.10. *Let $C_1, \ldots, C_k$ form an integral basis of G. Their undirected projection $\mathcal{B} := C'_1, \ldots, C'_k$ is a cycle basis for U.*

Proof. The result follows, if we can combine any undirected cycle $\tilde{C}$ of U from $\mathcal{B}$. Let C be a directed cycle of G with $C' = \tilde{C}$ as its projection. From the integrality property, we obtain

$$\gamma_C = \sum_{i=1}^{k} \lambda_i \gamma_i, \text{ with } \lambda_i \in \mathbb{Z}.$$

Since γ_C has only odd entries, the componentwise modulo 2 projection gives the desired result. ∎

Summarizing, figure 3.1 visualizes how cycle bases that are integral, undirected, or (strictly) fundamental are related. Apart from elementary cases, the examples in the appendices provide that none of the displayed regions is empty, except possibly for the shaded one.

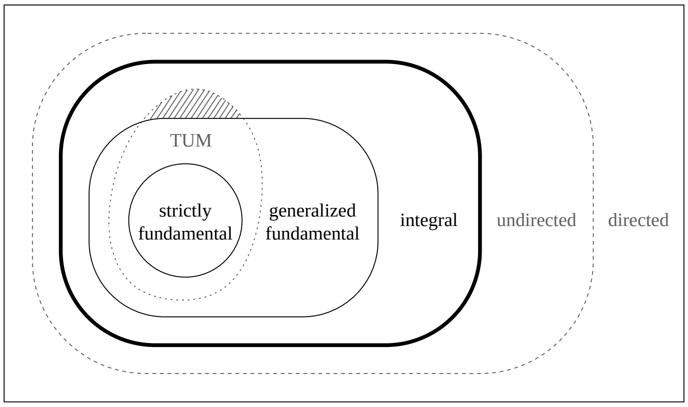

FIG. 3.1. *Map of Directed Cycle Bases*

4. Good Cycle Bases For Cyclic Railway Timetabling. The previous sections explained that it is sufficient to enforce the cycle periodicity constraints for all cycles in an integral cycle basis, and moreover, that fundamental cycle bases are integral. But a graph has many different cycle bases, and the question arises whether some are better than others for formulating and applying the CPF.

Suppose that we were to solve the CPF for the CRTP by brute force enumeration of the possibilities for the vector q. Let $W_C = b_C - a_C$ be the *width* of the cycle C. A cycle periodicity variable q_C can take $W_C + 1$ different values. For a cycle basis B, one sees that the vector $q = (q_1, \ldots, q_k)$ can take

$$W(B) = \prod_{C \in B} (W_C + 1) \tag{4.1}$$

different values. We call $W(B)$ the *width* of the cycle basis B. Therefore, brutely enumerating all possible values for q is done in a minimum number of iterations when using a cycle basis with minimum width. And also for sophisticated solution methods, such as Branch & Bound or Branch & Cut, it is sensible to formulate the CPF with a small or minimum width cycle basis. Therefore, we are interested in small width cycle bases for formulating the CPF.

4.1. Transforming the Cycle Basis Objective Function. The width of a cycle basis is a product and therefore a non-linear quantity. Moreover, the non-linear operation of rounding involved in computing a_C and b_C further obscures the construction of a small or minimum width cycle basis. First, in order to obtain a linear quantity for the width of a cycle basis, define

$$LW(B) = \log W(B) = \log \prod_{C \in B} (W_C + 1) = \sum_{C \in B} \log(W_C + 1). \tag{4.2}$$

Since the logarithm is a monotonous transformation, a minimum width cycle basis B^* also attains the minimum for the function LW.

Second, consider the impact of rounding in computing a_C and b_C, and thus in computing W_C. To that end, we forget about rounding for the moment, and consider the unrounded bounds a'_C and b'_C defined by

$$a'_C = \frac{1}{T}\left(\sum_{a \in C^+} \ell_a - \sum_{a \in C^-} u_a\right), \qquad b'_C = \frac{1}{T}\left(\sum_{a \in C^+} u_a - \sum_{a \in C^-} \ell_a\right).$$

For a cycle C we define its *unrounded width* W'_C as

$$\begin{aligned} W'_C = b'_C - a'_C &= \frac{1}{T}\left(\sum_{a \in C^+} u_a - \sum_{a \in C^-} \ell_a\right) - \frac{1}{T}\left(\sum_{a \in C^+} \ell_a - \sum_{a \in C^-} u_a\right) \\ &= \frac{1}{T}\left(\sum_{a \in C^+} (u_a - \ell_a) - \sum_{a \in C^-} (\ell_a - u_a)\right). \end{aligned}$$

That means that the direction of the arcs in C does not matter for the unrounded width W'_C of a cycle C. Moreover, W'_C is just a linear function of the arc widths $w_a = u_a - \ell_a$. The gap between the unrounded and rounded width of a cycle C equals

$$W'_C - W_C = \left(b'_C - \lfloor b'_C \rfloor\right) + \left(\lceil a'_C \rceil - a'_C\right).$$

Using that, for some $y \in \mathbb{R}, 0 \leq y - \lfloor y \rfloor < 1$ and $0 \leq \lceil y \rceil - y < 1$, we obtain

$$0 \leq W'_C - W_C < 2.$$

So, as a heuristic for approximating $LW(B)$, and thus $W(B)$, we could consider minimizing the following cycle basis objective function

$$LW'(B) = \sum_{C \in B} \log(W'_C + 1) = \sum_{C \in B} \log(1 + \sum_{a \in A} w_a). \tag{4.3}$$

And, since the direction of the arcs does not matter in this function, $LW'(B)$ may also be defined on the underlying undirected graph U of G. In that case, minimizing $LW'(B)$ is very similar to the so-called *minimum cycle basis problem* for undirected graphs. As the complexity of finding a minimal fundamental cycle basis or even finding a minimal integral cycle basis is unknown to the authors, the next section describes minimum cycle bases of undirected graphs.

4.2. Minimum Cycle Bases. Consider the problem of finding a so-called minimum cycle basis in an undirected graph.

DEFINITION 4.1. *Given an undirected graph $U = (N, E)$ with edge weights w_e for all $e \in E$, a* minimum cycle basis *of G is a cycle basis B^* that minimizes the cycle basis* weight

$$\sum_{C \in B^*} \sum_{e \in C} w_e. \tag{4.4}$$

A minimum (strictly) fundamental cycle basis *is a minimum cycle basis among the (strictly) fundamental cycle bases.*

Many authors studied minimum cycle bases. Horton (1987) has developed an $O(m^3 n)$ algorithm for constructing a minimum cycle basis. However, the previous section showed that we need an integral cycle basis for the CPF, and Horton's algorithm may return a non-integral cycle basis. Deo et al. (1982) proved that the problem of finding a minimum strictly fundamental cycle basis[3] for the unit edge weight case is $\mathcal{NP}$-complete.

4.3. Minimum Cycle Bases and the CPF. Since the minimum cycle basis problem has already been studied in literature, and algorithms are available, we propose to use minimum cycle basis algorithms as heuristics for constructing small width cycle bases for formulating the CPF. However, such an algorithm may yield a non-integral cycle basis, so the solution should be checked carefully afterwards. Alternatively, heuristics for constructing minimum strictly fundamental cycle bases can be used. Theorem 3.8 shows that these cycle bases are integral.

We are aware of loosing information when using minimum (strictly fundamental) cycle bases for approximating the minimum of $W(B)$ over all integral cycle bases. The main inexactness stems from not considering the rounding for computing the width of a cycle. To illustrate the inexactness of applying the unrounded width, consider the constraint graph $K_n, n \geq 7$ with $T = 10$, and the infeasible time windows $[l_a, u_a] = [3, 4]$ for all arcs $a \in A$. The shortest infeasible cycles have seven arcs. But a minimum cycle basis for the underlying undirected graph U, with respect to the edge weights $w_e = w_a = 1$, will only consist of triangles, which are all feasible.

Moreover, a minimum cycle basis for an undirected graph does not have to be fundamental. Leydold and Stadler (1998) provide an example of this phenomenon on K_9, and we propose the smallest possible example, defined on K_8, in Appendix C.

Finally, heuristics for the minimum strictly fundamental cycle basis problem that occasionally only take into account the edge weights may lead to quite poor results. Consider K_n with $w_e = 1$ for all $e \in E$. A minimum strictly fundamental cycle basis consists of triangles, and corresponds to a rooted star tree in G. This cycle basis has total weight $3k = \frac{3}{2}(n-1)(n-2)$, because each of the $\binom{n-1}{2}$ chords induces a triangle. But as the heuristics to analyze do not take into account the degree of the nodes in the tree, they will not be able to distinguish a star tree from a Hamiltonian Path. For the latter, we obtain the total weight by counting how may non-tree edges do close a cycle of a given length:

$$\sum_{i=3}^{n} (n - i + 1) \cdot i = \frac{1}{6}(n-1)(n-2)(n+6).$$

[3]Deo et al. use the term 'minimum-length fundamental cycle set'.

Here, the index i denotes the lengths of the cycles. This shows that such heuristics do not even possess constant performance ratios.

5. Benefit of Cycles in Solving CRTP. In the previous sections, we discussed the benefit of cycles when formulating the CRTP. In this section, we shortly cite some important results showing how we can profit from cycles when solving CRTP instances, too.

In section 2, we mentioned that the CRTP becomes easy, after all the integer variables have been fixed. This is why we define the polyhedron $\mathcal{Z}$ to be the convex hull of the integral vectors p that permit feasible schedules (v, p). Together with Nachtigall, Lindner (2000) discovered that cycles are necessary in order to express any valid inequality of $\mathcal{Z}$.

THEOREM 5.1 (Lindner, 2000). *$\varphi^t p \geq \varphi_0$ is a valid inequality for $\mathcal{Z}$, if and only if φ is a circulation and $\varphi_0 \leq \min\{\varphi^t p \,|\, p \in \mathcal{Z}\}$.*

The cycle inequalities (2.5) are an important special case of this theorem. But they do not form the complete description of $\mathcal{Z}$, because sometimes they are not tight. There are constraint graphs in that the original lower bound a_C of a cycle inequality still has to be incremented in order to attain the above minimum (Lindner, 2000).

For solving the (mixed) integer linear program (2.4) in a Branch & Cut context, we want to have further valid inequalities at hand. Nachtigall (1996a) came up with another class of valid inequalities, that are – again – defined for every cycle of the constraint graph. As originally they were expressed in terms of unconstrained *slack variables* $\tilde{x} := x - l$, we reformulate them in the following theorem. To that aim, for a cycle C of the constraint graph we propose the follwing definitions:

$$\begin{array}{llll} a^+ = \sum_{a \in C^+} l_a & a^- = \sum_{a \in C^-} l_a & \alpha = a^- - a^+ & a = \alpha \bmod T, \\ b^+ = \sum_{a \in C^+} u_a & b^- = \sum_{a \in C^-} u_a & \beta = b^- - b^+ & b = \beta \bmod T. \end{array}$$

THEOREM 5.2 (Nachtigall, 1996a). *Consider a CRTP given by its CPF* (2.4). *Let C be a cycle of the constraint graph. If $b > 0$ and $a > 0$, then the following* change cycle inequalities *are valid:*

$$\sum_{a \in C^-} x_a \geq -\frac{T - a}{a} \sum_{a \in C^+} x_a + \left(\frac{a^+}{a} + \left\lceil \frac{\alpha}{T} \right\rceil \right) T \tag{5.1}$$

$$\sum_{a \in C^-} x_a \leq -\frac{b}{T - b} \sum_{a \in C^+} x_a + \left(\frac{b^+}{b} - \left\lceil \frac{\beta}{T} \right\rceil \right) T. \tag{5.2}$$

For cycles C with $C^- \neq \emptyset$, Peeters (2003) gives a visualization of both, cycle inequalities and change cycle inequalities within one common chart, see figure 5.1. The lines l_1 and l_2 depict the borders of the halfspaces defined by inequalities (5.1) and (5.2). The notation for the cycle inequalities (2.5) is just as in theorem 2.2, together with equality (2.3).

Empirical studies performed by Liebchen (1998) confirm what figure 5.1 suggests. Neither change cycle inequalities nor cycle inequalities (2.5) dominate the other class in practice. Rather, both cycle cuts are complementary.

Rests the question which inequalities to choose from the huge stock of three inequalities per cycle. Determining the sequence of trains on a track frequented by several lines is one of the major decisions to be made in cyclic timetabling. Liebchen and Peeters (2002) investigated the subgraphs, i.e. cliques, that ensure such safety distances, in more detail.

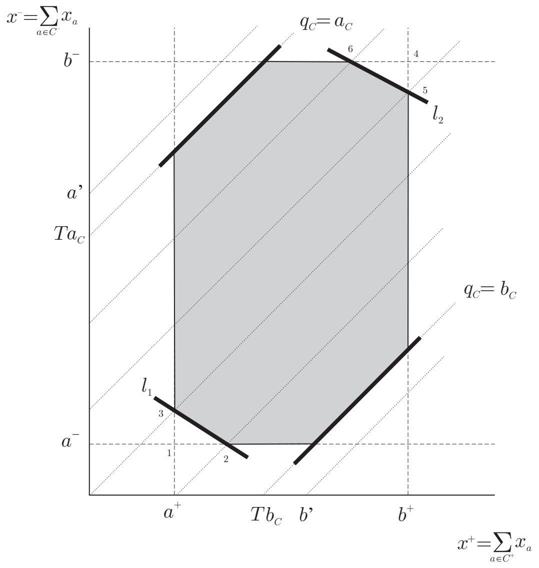

FIG. 5.1. *Cycle Inequalities and Change Cycle Inequalities*

Consider the events of n lines to be separated and, for notational convenience, assume the cycle basis to stem from a star tree H, hence being strictly fundamental. Then, due to the cycle inequalities, for each of the $\binom{n-1}{2}$ chords, its integer variable may be boxed by zero and one. But these box constraints only give a poor relaxation $\mathcal{B} := \{0,1\}^{\binom{n-1}{2}}$ of the polyhedron $\mathcal{Z}$, where we extend an integral vector z to p by setting $p_a = 0$ for $a \in H$.

In fact, we are interested in only $(n-1)!$ integral vectors corresponding to a permutation, or sequence, of the n lines to be separated. Fortunately, for any integral vector in $\mathcal{B} \setminus \mathcal{Z}$, there is a cycle inequality violated for some non-basic triangle. Hence, one may add the $2\binom{n-1}{3} \in \mathcal{O}(n^3)$ cycle inequalities for every triangle explicitly, in order to cut off the $2^{\binom{n-1}{2}} - (n-1)!$ infeasible integral vectors from the initial box polytope $\mathcal{B}$. As the integer variables can also be interpreted as decision variables for the Linear Ordering Problem (Grötschel et al., 1984), this is generalized by the PESP.

THEOREM 5.3 (Liebchen and Peeters, 2002). *The Linear Ordering Problem is polynomially reducible to the problem of minimizing a linear objective function over an instance of the Periodic Event Scheduling Problem, with fixed interval time T.*

6. Conclusions. We did introduce the concept of integral cycle bases. They are an appropriate tool for characterizing periodic tensions. Although it is an advantage to be no more limited to strictly fundamental cycle bases in formulating the CRTP, additional benefit would immediately arise from a better understanding of (minimal) integral or fundamental cycle bases. Two major questions to be answered are:

- What is the complexity of computing a minimal integral cycle basis of a directed graph?
- Does every graph have some minimal cycle basis that is fundamental?

Moreover, cycles of the constraint graph play an essential role in Branch & Cut approaches for solving the CRTP. For every cycle, there are four valid inequalities known, and the cycles in subgraphs that result from separation constraints are a promising source for cuts. The latter is closely related to a new proof for the $\mathcal{NP}$-

completeness of the Periodic Event Scheduling Problem.

7. Acknowledgement. We appreciate the worthwile hints of Sabine Cornelsen.

Appendix A. Example of a Non-integral Cycle Basis.

Section 3.2 mentioned that a solution to (2.4) may violate the cycle periodicity property for non-basic cycles, when the cycle periodicity constraints are required only for the cycle in a non-integral cycle basis. Consequently, the solution may yield a timetable that violates some constraints. The directed graph G in figure A.1 provides an example of this situation. Hartvigsen and Zemel (1989) presented this graph to prove a theorem stating under which conditions every cycle basis of an undirected graph is fundamental.

The undirected counterparts of the four cycles $C_1, \ldots, C_4$ form a basis for the underlying undirected graph U over GF(2), since $k = 4$, and no subset of the cycles sums up to zero modulo 2. By theorem 3.9, it follows that $B = \{C_1, \ldots, C_4\}$ is a basis for G. Note that B is not a fundamental cycle basis, since each arc in G appears in at least two cycles of the basis. It is therefore not possible to re-order the cycles $C_1, \ldots, C_4$ such that $C_i \setminus (C_{i-1} \cup \ldots \cup C_1) \neq \emptyset$ for $i = 2, 3, 4$.

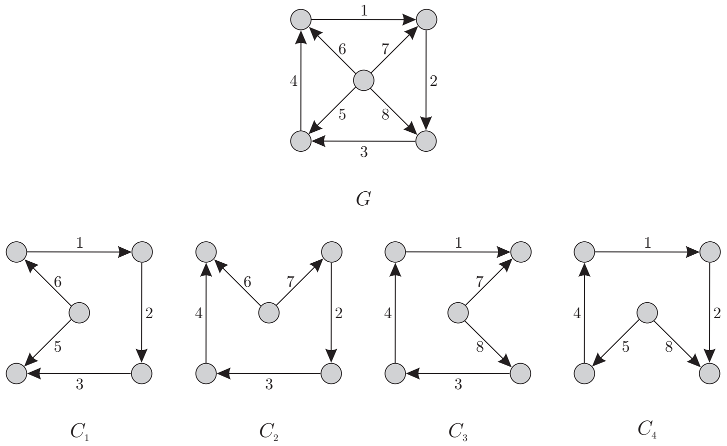

FIG. A.1. *Directed graph G with cycle basis $B = \{C_1, \ldots, C_4\}$*

The cycle matrix corresponding to the cycle basis B is the following:

$$\Gamma_B = \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \begin{bmatrix} 1 & 1 & 1 & 0 & -1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & -1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & -1 \end{bmatrix} \end{array} \begin{array}{c} \\ \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \gamma_4 \end{array}$$

Now consider the non-basic cycles C_5, C_6, C_7 in figure A.2, with their corresponding incidence vectors:

$$\gamma_5 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{bmatrix}$$
$$\gamma_6 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & -1 \end{bmatrix}$$
$$\gamma_7 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

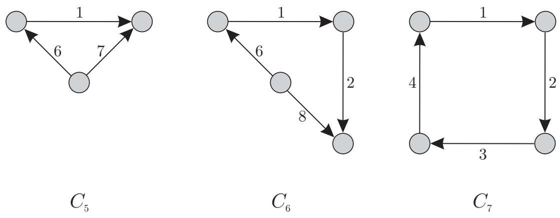

FIG. A.2. *Non-basic cycles C_5, C_6, C_7*

The vectors $\gamma_5, \gamma_6, \gamma_7$ can be expressed in the basis incidence vectors as follows

$$\begin{aligned}\gamma_5 &= \tfrac{1}{3}\gamma_1 - \tfrac{2}{3}\gamma_2 + \tfrac{1}{3}\gamma_3 + \tfrac{1}{3}\gamma_4,\\ \gamma_6 &= \tfrac{2}{3}\gamma_1 - \tfrac{1}{3}\gamma_2 - \tfrac{1}{3}\gamma_3 + \tfrac{2}{3}\gamma_4,\\ \gamma_7 &= \tfrac{1}{3}\gamma_1 + \tfrac{1}{3}\gamma_2 + \tfrac{1}{3}\gamma_3 + \tfrac{1}{3}\gamma_4.\end{aligned}$$

Since B is a basis, these expressions are unique. It follows that B is non-integral. Note that the three cycles C_5, C_6, C_7 represent the structure of *all* cycles in G that are not in the cycle basis, and so *every* non-basic cycle in G is a non-integer linear combination of the basic cycles. Changing the direction of a cycle or arc does not disturb the result, since the former means changing the sign of a row, and the latter means changing the sign of a column.

To illustrate that this may lead to a false solution, consider the PESP instance in figure A.3(a), with $T = 60$. Figure A.3(b) shows the solution

$$\bar{x} = \begin{bmatrix}\bar{x}_1, \dots, \bar{x}_8\end{bmatrix} = \begin{bmatrix}20 & 20 & 20 & 20 & 15 & 15 & 15 & 15\end{bmatrix},$$

which is feasible with respect to the cycle periodicity constraints

$$\sum_{a\in C^+} \bar{x}_a - \sum_{a\in C^-} \bar{x}_a = 60 \text{ for all } C \in B, \tag{A.1}$$

where the cycle basis B still consists of the cycles $C_1, \dots, C_4$. However, for the non-basic cycles C_5, C_6, C_7, we have respectively

$$\bar{x}_1 + \bar{x}_6 - \bar{x}_7 = 20, \tag{A.2}$$

$$\bar{x}_1 + \bar{x}_2 + \bar{x}_6 - \bar{x}_8 = 40, \tag{A.3}$$

$$\bar{x}_1 + \bar{x}_2 + \bar{x}_3 + \bar{x}_4 = 80. \tag{A.4}$$

Since $T = 60$, these cycles clearly do not satisfy the cycle periodicity property. Finally, consider figure A.3(c), that tries to construct a node potential solution for the periodic tension solution $\bar{x}$. Choosing the value 0 for the center node, the corner nodes each have to take the value 15 because of the value $\bar{x}_a = 15$ for the bold tree arcs $a = 5, \dots, 8$. But clearly, this node potential does not comply with the periodic tensions $\bar{x}_a = 20$ for the arcs $a = 1, \dots, 4$.

Appendix B. Directed Cycle Basis Not Inducing Undirected Basis. Figure B.1 provides an example of a directed graph G, and a cycle basis $B = \{C_1, C_2, C_3\}$ that does *not* form a basis for the underlying undirected graph U. The cycle matrix Γ_B is displayed below.

First, through inspection of the possible cases, it can be seen that none of the three cycles C_1, C_2, C_3 can be expressed as a linear combination of the other two, so

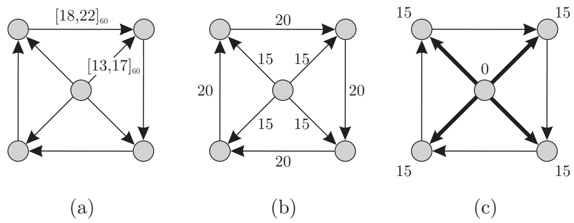

FIG. A.3. *(a) Infeasible PESP instance, (b) feasible solution $\bar{x}$ for* (A.1)*, (c) impossible to construct feasible potential.*

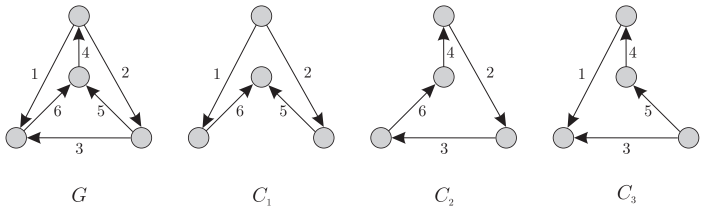

FIG. B.1. $B = \{C_1, \ldots, C_3\}$ *forms a basis for* G*, but not for the underlying undirected graph* U

the cycles form a cycle basis of G. Second, when considering the three cycles in the underlying undirected graph G, actually *each* cycle is equal to the sum of the other modulo 2, so C_1, C_2, C_3 do not form a basis for U.

$$\Gamma_B = \begin{array}{c} \begin{array}{cccccc} \scriptstyle 1 & \scriptstyle 2 & \scriptstyle 3 & \scriptstyle 4 & \scriptstyle 5 & \scriptstyle 6 \end{array} \\ \begin{bmatrix} 1 & 1 & 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 0 & 1 & -1 \\ 1 & 0 & 1 & -1 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{array}$$

This obviously is the smallest possible example.

The smallest set of directed cycles that form a minimum directed cycle basis, but that does not impose a cycle basis for the underlying undirected simple graph is

(1, 2, 3) (1, 2, 4) (1, 3, 5) (1, 4, 6) (1, 5, 6)
(2, 3, 6) (2, 4, 5) (2, 5, 6) (3, 4, 5) (3, 4, 6)

on the directed K_6. On one hand, the set only contains triangles, hence it is minimum. And as there are 10×10 submatrices of the cycle matrix that have non-zero determinant, it is indeed a directed cycle basis. On the other hand, as every edge is hit exactly twice, it cannot be an undirected cycle basis. The example is node-minimal, because we have to cover every arc at least twice, but a cycle basis consisting only of triangles, covers all the arcs together only $3k = 3(m-n+1)$ times. From $2m \leq 3(m-n+1)$ we immedialtely deduce $m \geq 3n-3$, which for simple graphs can only be true for $n \geq 6$.

Appendix C. Minimum Cycle Basis Not Being Fundamental. The fact that there are minimum undirected cycle bases that are not fundamental has been proven by Leydold and Stadler (1998). They gave an example on the complete graph on nine vertices K_9. We do propose a smaller example, on K_8:

{1, 2, 3} {1, 2, 4} {1, 2, 5} {1, 3, 4} {1, 3, 5} {1, 4, 6} {1, 5, 7}
{1, 6, 8} {1, 7, 8} {2, 3, 6} {2, 3, 7} {2, 4, 7} {2, 5, 8} {2, 6, 8}
{3, 4, 8} {3, 5, 6} {3, 7, 8} {4, 5, 6} {4, 5, 8} {4, 6, 7} {5, 6, 7}

This is a minimum cycle basis, because it only contains the smallest possible cycles, and it leads to a matrix of full row rank over GF(2). Nevertheless, it is not fundamental, as every edge is contained in at least two triangles.

Moreover, K_8 is a node-minimal graph on that one may find such an example. Theoretically, K_7 could contain 15 triangles that are linearly independent and that do cover every edge by at least two basic cycles. Simple counting tells us that then there have to be three edges that will be be hit precisely three times. But if they did not constitute a triangle themselves, there would be vertices of odd degree, which is impossible in a union of cycles. Hence, we just have to add the basic cycles that do *not* appear in the linear combination for representing the triangle containing the edges that are hit three times, in order to come up with a non-trivial combination of the null vector, proving that K_7 cannot provide such an example.

Appendix D. Fundamental Cycle Basis without TUM Cycle Matrix. Figure D.1 provides an example of a directed graph G, and a fundamental cycle basis $B = \{C_1, C_2, C_3\}$ that does *not* lead to a totally unimodular cycle matrix.

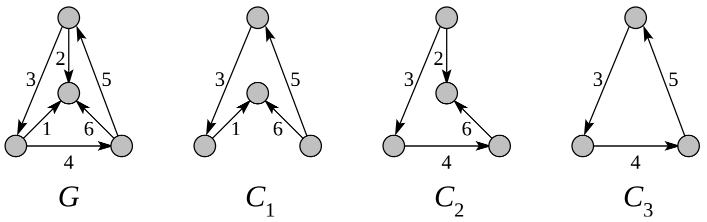

FIG. D.1. *Fundamental cycle basis $B = \{C_1, C_2, C_3\}$, not having totally unimodular cycle matrix*

The cycle matrix Γ_B is displayed below.

$$\Gamma_B = \begin{array}{c} \begin{array}{cccccc} \scriptstyle 1 & \scriptstyle 2 & \scriptstyle 3 & \scriptstyle 4 & \scriptstyle 5 & \scriptstyle 6 \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & -1 \\ 0 & 1 & -1 & -1 & 0 & -1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix} \end{array} \quad \begin{array}{c} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{array}$$

As the three first columns form an upper triangular matrix, B is fundamental. But as the 2×2-submatrix of columns 3 and 6, and rows γ_1 and γ_2, has determinant $-2 \notin \{0, \pm 1\}$, Γ_B is not totally unimodular.

Appendix E.

BERGE, C. (1962). *The Theory of Graphs and its Applications*. John Wiley & Sons, New York.

BOLLOBÁS, B. (1998). *Modern Graph Theory*, volume 184 of *Graduate Texts in Mathematics*. Springer, Berlin.

BUSSIECK, M., WINTER, T., ZIMMERMANN, U. (1997). *Discrete optimization in public rail transportation*. Mathematical Programming, 79(3):415–444.

DEO, N. (1982). *Graph Theory with Applications to Engineering and Computer Science*. Prentice–Hall Series in Automatic Computation. Prentice–Hall, Englewood Cliffs.

DEO, N., PRABHU, G.M., KRISHNAMOORTHY, M.S. (1982). *Algorithms for generating fundamental cycles in a graph*. ACM Transactions on Mathematical Software, 8:26–42.

GRÖTSCHEL, M., JÜNGER, M., REINELT, G. (1984). *A cutting plane algorithm for the linear ordering problem*. Operations Research, 32(6):1195–1220.

Hartvigsen, D., Zemel, E. (1989). *Is every cycle basis fundamental?*. Journal of Graph Theory, 13(1):117–137.

Hooghiemstra, J., Kroon, L., Odijk, M., Salomon, M., Zwaneveld, P. (1999). *Decision support systems support the search for win-win solutions in railway network design*. Interfaces, 29(2):15–32.

Horton, J. (1987). *A polynomial-time algorithm to find the shortest cycle basis of a graph*. SIAM Journal on Computing, 16(2):358–366.

Krista, M. (1996). *Verfahren zur Fahrplanoptimierung dargestellt am Beispiel der Synchronzeiten (Methods for timetable optimization illustrated by synchronous times)*. Ph.D. thesis, Technical University Braunschweig, Braunschweig, Germany. In German.

Leydold, J., Stadler, P. (1998). *Minimal cycle bases of outerplanar graphs*. The Electronic Journal of Combinatorics, 5:R16.

Liebchen, C. (1998). *Optimierungsverfahren zur Erstellung von Taktfahrplänen (Optimization methods for constructing periodic timetables)*. Master's thesis, Technical University Berlin, Berlin, Germany. In German.

Liebchen, C., Möhring, R.H. (2002). *A case study in periodic timetabling*. In D. Wagner (editor), *Electronic Notes in Theoretical Computer Science*, volume 66. Elsevier.

Liebchen, C., Peeters, L. (2002). *Some practical aspects of periodic timetabling*. In P. Chamoni, R. Leisten, A. Martin, J. Minnemann, H. Stadtler (editors), *Operations Research 2001*. Springer.

Lindner, T. (2000). *Train schedule optimization in public transport*. Ph.D. thesis, TU Braunschweig, Braunschweig, Germany.

Nachtigall, K. (1994). *A branch and cut approach for periodic network programming*. Technical Report 29, Hildesheimer Informatik-Berichte.

——— (1996a). *Cutting planes for a polyhedron associated with a periodic network*. Technical Report 17, DLR Interner Bericht.

——— (1996b). *Periodic network optimization with different arc frequencies*. Discrete Applied Mathematics, 69:1–17.

Nachtigall, K., Voget, S. (1996). *A genetic algorithm approach to periodic railway synchronization*. Computers and Operations Research, 23(5):453–463.

——— (1997). *Minimizing waiting times in integrated fixed interval timetables by upgrading railway tracks*. European Journal of Operational Research, 103(3):610–627.

Odijk, M. (1996). *A constraint generation algorithm for the construction of periodic railway timetables*. Transportation Research, 30(6):455–464.

——— (1997). *Railway timetable generation*. Ph.D. thesis, Delft University of Technology, Delft, The Netherlands.

Peeters, L. (2003). *Cyclic railway timetable optimization*. Ph.D. thesis, Erasmus University Rotterdam, Rotterdam, The Netherlands. To appear.

Schrijver, A. (1998). *Theory of Linear and Integer Programming*. John Wiley & Sons, New York, second edition.

Schrijver, A., Steenbeek, A. (1993). *Spoorwegdienstregelingontwikkeling (Timetable construction)*. Technical report, CWI Center for Mathematics and Computer Science, Amsterdam, The Netherlands. In Dutch.

Serafini, P., Ukovich, W. (1989). *A mathematical model for periodic event scheduling problems*. SIAM Journal on Discrete Mathematics, 2(4):550–581.

Stockmeyer, L.J. (1973). *Planar 3-colorability is polynomial complete*. ACM

SIGACT News, 5:19–25.

VOORHOEVE, M. (1993). *Rail scheduling with discrete sets*. Technical report, Eindhoven University of Technology, Eindhoven, The Netherlands.

WHITNEY, H. (1935). *The abstract properties of linear dependence*. American Journal on Mathematics, 57:509–533.

Reports from the group

"Combinatorial Optimization and Graph Algorithms"

of the Department of Mathematics, TU Berlin

762/2002 *Ekkehard Köhler and Katharina Langkau and Martin Skutella:* Time-Expanded Graphs for Flow-Dependent Transit Times

761/2002 *Christian Liebchen and Leon Peeters:* On Cyclic Timetabling and Cycles in Graphs

752/2002 *Ekkehard Köhler and Rolf H. Möhring and Martin Skutella:* Traffic Networks and Flows Over Time

739/2002 *Georg Baier and Ekkehard Köhler and Martin Skutella:* On the k-splittable Flow Problem

736/2002 *Christian Liebchen and Rolf H. Möhring:* A Case Study in Periodic Timetabling

723/2001 *Berit Johannes:* Scheduling Parallel Jobs to Minimize Makespan

716/2001 *Christian Liebchen:* The Periodic Assignment Problem (PAP) May Be Solved Greedily

711/2001 *Esther M. Arkin, Michael A. Bender, Sándor P. Fekete, Joseph S. B. Mitchell, and Martin Skutella:* The Freeze-Tag Problem: How to Wake Up a Swarm of Robots

710/2001 *Esther M. Arkin, Sándor P. Fekete, and Joseph S. B. Mitchell:* Algorithms for Manufacturing Paperclips and Sheet Metal Structures

705/2000 *Ekkehard Köhler:* Recognizing Graphs without Asteroidal Triples

704/2000 *Ekkehard Köhler:* AT-free, coAT-free Graphs and AT-free Posets

702/2000 *Frederik Stork:* Branch-and-Bound Algorithms for Stochastic Resource-Constrained Project Scheduling

700/2000 *Rolf H. Möhring:* Scheduling under uncertainty: Bounding the makespan distribution

698/2000 *Sándor P. Fekete, Ekkehard Köhler, and Jürgen Teich:* More-dimensional packing with order constraints

697/2000 *Sándor P. Fekete, Ekkehard Köhler, and Jürgen Teich:* Extending partial suborders and implication classes

696/2000 *Sándor P. Fekete, Ekkehard Köhler, and Jürgen Teich:* Optimal FPGA module placement with temporal precedence constraints

695/2000 *Sándor P. Fekete, Henk Meijer, André Rohe, and Walter Tietze:* Solving a "hard" problem to approximate an "easy" one: heuristics for maximum matchings and maximum Traveling Salesman Problems

694/2000 *Esther M. Arkin, Sándor P. Fekete, Ferran Hurtado, Joseph S. B. Mitchell, Marc Noy, Vera Sacristán and Saurabh Sethia:* On the reflexivity of point sets

693/2000 *Frederik Stork and Marc Uetz:* On the representation of resource constraints in project scheduling

691/2000 *Martin Skutella and Marc Uetz:* Scheduling precedence constrained jobs with stochastic processing times on parallel machines

689/2000 *Rolf H. Möhring, Martin Skutella, and Frederik Stork:* Scheduling with AND/OR precedence constraints

685/2000 *Martin Skutella:* Approximating the single source unsplittable min-cost flow problem

684/2000 *Han Hoogeveen, Martin Skutella, and Gerhard J. Woeginger:* Preemptive scheduling with rejection

683/2000 *Martin Skutella:* Convex quadratic and semidefinite programming relaxations in Scheduling

682/2000 *Rolf H. Möhring and Marc Uetz:* Scheduling scarce resources in chemical engineering

681/2000 *Rolf H. Möhring:* Scheduling under uncertainty: optimizing against a randomizing adversary

680/2000 *Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz:* Solving project scheduling problems by minimum cut computations (Journal version for the previous Reports 620 and 661)

674/2000 *Esther M. Arkin, Michael A. Bender, Erik D. Demaine, Sándor P. Fekete, Joseph S. B. Mitchell, and Saurabh Sethia:* Optimal covering tours with turn costs

669/2000 *Michael Naatz:* A note on a question of C. D. Savage

667/2000 *Sándor P. Fekete and Henk Meijer:* On geometric maximum weight cliques

666/2000 *Sándor P. Fekete, Joseph S. B. Mitchell, and Karin Weinbrecht:* On the continuous Weber and k-median problems

664/2000 *Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz:* On project scheduling with irregular starting time costs

661/2000 *Frederik Stork and Marc Uetz:* Resource-constrained project scheduling: from a Lagrangian relaxation to competitive solutions

Reports may be requested from:

Sekretariat MA 6–1
Fakultt II – Institut fr Mathematik
TU Berlin
Straße des 17. Juni 136
D-10623 Berlin – Germany

e-mail: klink@math.TU-Berlin.DE

Reports are also available in various formats from

`http://www.math.tu-berlin.de/coga/publications/techreports/`

and via anonymous ftp as

`ftp://ftp.math.tu-berlin.de/pub/Preprints/combi/Report-`*number*`-`*year*`.ps`