Finding Short Integral Cycle Bases for Cyclic Timetabling*

Christian Liebchen

TU Berlin, Institut für Mathematik, Sekr. MA 6-1
Straße des 17. Juni 136, D-10623 Berlin, Germany
liebchen@math.tu-berlin.de

Abstract. Cyclic timetabling for public transportation companies is usually modeled by the periodic event scheduling problem. To obtain a mixed-integer programming formulation, artificial integer variables have to be introduced. There are many ways to define these integer variables.
We show that the minimal number of integer variables required to encode an instance is achieved by introducing an integer variable for each element of some integral cycle basis of the directed graph $D = (V, A)$ defining the periodic event scheduling problem. Here, integral means that every oriented cycle can be expressed as an *integer* linear combination.
The solution times for the originating application vary extremely with different integral cycle bases. Our computational studies show that the width of integral cycle bases is a good empirical measure for the solution time of the MIP. Integral cycle bases permit a much wider choice than the standard approach, in which integer variables are associated with the co-tree arcs of some spanning tree. To formulate better solvable integer programs, we present algorithms that construct good integral cycle bases. To that end, we investigate subsets and supersets of the set of integral cycle bases. This gives rise to both, a compact classification of directed cycle bases and notable reductions of running times for cyclic timetabling.

1 Introduction and Scope

Cycle bases play an important role in various applications. Recent investigations cover ring perception in chemical structures ([8]) and the design and analysis of electric networks ([3]). Cyclic timetabling shares with these applications that the construction of a good cycle basis is an important preprocessing step to improve solution methods for real world problems.

Since the pioneering work of Serafini and Ukovich[23], the construction of periodic timetables for public transportation companies, or cyclic timetabling for short, is usually modeled as a periodic event scheduling problem (PESP). For an exhaustive presentation of practical requirements that the PESP is able to meet, we refer to Krista[12]. The feasibility problem has been shown to be $\mathcal{NP}$-complete, by reductions from Hamiltonian Cycle ([23] and [18]) or Coloring ([20]). The minimization problem with a linear objective has been shown to be $\mathcal{NP}$-hard by a reduction from Linear Ordering ([16]). We want to solve PESP instances by using the mixed integer solver of CPLEX© [5].

* Supported by the DFG Research Center "Mathematics for key technologies" in Berlin

G. Di Battista and U. Zwick (Eds.): ESA 2003, LNCS 2832, pp. 715–726, 2003.

Related Work. The performance of implicit enumeration algorithms for mixed integer programming can be improved by reducing the number of integer variables. Already Serafini and Ukovich detected that there is no need to introduce an integer variable for every arc of the directed constraint graph. Rather, one can restrict the integer variables to those that correspond to the co-tree arcs of a spanning tree. These arcs can be interpreted to be the representatives of a strictly fundamental cycle basis.

Nachtigall[17] profited from the spanning tree approach when switching to a tension-based problem formulation. Notice that our results on integral cycle bases apply to that tension-perspective as well. Odijk[20] provided box constraints for the remaining integer variables. Hereby, it becomes possible to quantify the difference between cycle bases. But the implied objective function for finding a short integral cycle basis is bulky. De Pina[21] observed that a cycle basis that minimizes a much simpler function also minimizes our original objective. What remains to solve is a variant of the minimal cycle basis problem.

Contribution and Scope. We show that the width of a cycle basis is highly correlated with the solution time of the MIP solver. Thus, it serves as a good empirical measure for the run time and provides a way to speed up the solver by choosing a good basis.

Hence, in order to supply MIP solvers with promising problem formulations, we want to compute short directed cycle bases which are suitable for expressing PESP instances. But there is a certain dilemma when analyzing the two most popular types of directed cycle bases: On the one hand, there are directed cycle bases that induce undirected cycle bases. For these, we can minimize a linear objective function efficiently (Horton[11]). But, contrary to a claim of de Pina[21], undirected cycle bases unfortunately are *not* applicable to cyclic timetabling in general – we give a counter-example. On the other hand, strictly fundamental cycle bases form a feasible choice. But for them, minimization is $\mathcal{NP}$-hard (Deo et al.[7]).

To cope with this dilemma, we investigate if there is a class of cycle bases lying in *between* general undirected cycle bases and strictly fundamental cycle bases, hopefully combining both, good algorithmic behavior and the potential to express PESP instances. To that end, we will present a compact classification of directed cycle bases. Efficient characterizations will be based on properties of the corresponding cycle matrices, e.g. its determinant, which we establish to be well-defined. This allows a natural definition of the *determinant of a directed cycle basis*.

An important special class are integral cycle bases. They are the most general structure when limiting a PESP instance to $|A|-|V|+1$ integer variables. But the complexity of minimizing a linear objective over the integral cycle bases is unknown to the author.

The computational results provided in Section 6 show the enormous benefit of generalizing the spanning tree approach to integral cycle bases for the originating application of cyclic timetabling. These results point out the need of deeper insights into integral cycle bases and related structures. Some open problems are stated at the end.

2 Periodic Scheduling and Short Cycle Bases

An instance of the Periodic Event Scheduling Problem (PESP) consists of a directed constraint graph $D = (V, A, \ell, u)$, where ℓ and u are vectors of lower and upper time bounds for the arcs, together with a period time T of the transportation network. A solution of a PESP instance is a node potential $\pi : V \to [0, T)$—which is a time vector for the periodically recurring departure/arrival events within the public transportation network—fulfilling periodic constraints of the form $(\pi_j - \pi_i - \ell_{ij}) \bmod T \leq u_{ij} - \ell_{ij}$. We reformulate the $\bmod$ operator by introducing artificial integer variables p_{ij},

$$\ell_{ij} \leq \pi_j - \pi_i + p_{ij} T \leq u_{ij},\ (i,j) \in A. \tag{1}$$

Our computational results will show that the running times of a mixed-integer solver on instances of cyclic timetabling correlate with the volume of the polytope spanned by box constraints provided for the integer variables. Formulation (1) permits three values $p_a \in \{0, 1, 2\}$ for $a \in A$ in general,[1] even with scaling to $0 \leq \ell_{ij} < T$.

Serafini and Ukovich observed that the above problem formulation may be simplified by eliminating $|V| - 1$ integer variables that correspond to the arcs a of some spanning tree H, when relaxing π to be some real vector. Formally, we just fix $p_a := 0$ for $a \in H$. Then, in general, the remaining integer variables may take more than three values. For example, think of the directed cycle on n arcs, with $\ell \equiv 0$ and $u \equiv T - \frac{1}{n}$, as constraint graph. With $\pi = \mathbf{0}$, the integer variable of every arc will be zero. But $\pi_i = (i-1) \cdot (T - \frac{1}{n}),\ i = 1, \ldots, n$ would be a feasible solution as well, implying $p_{n1} = n - 1$ for the only integer variable that we did not fix to zero. Fortunately, Theorem 1 provides box constraints for the remaining integer variables.

Theorem 1 (Odijk[20]). *A PESP instance defined by the constraint graph $D = (V, A, \ell, u)$ and a period time T is feasible if and only if there exists an integer vector $p \in \mathbb{Z}^{|A|}$ satisfying the* cycle inequalities

$$a_C \leq \sum_{a \in C^+} p_a - \sum_{a \in C^-} p_a \leq b_C, \tag{2}$$

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

$$a_C = \left\lceil \frac{1}{T} \left(\sum_{a \in C^+} \ell_a - \sum_{a \in C^-} u_a \right) \right\rceil, \quad b_C = \left\lfloor \frac{1}{T} \left(\sum_{a \in C^+} u_a - \sum_{a \in C^-} \ell_a \right) \right\rfloor, \tag{3}$$

and C^+ and C^- denote the sets of arcs that, for a fixed orientation of the cycle, are traversed forwardly resp. backwardly.

For any co-tree arc a, the box constraints for p_a can be derived by applying the cycle inequalities (2) to the unique oriented cycle in $H \cup \{a\}$.

Directed Cycle Bases and Undirected Cycle Bases. Let $D = (V, A)$ denote a connected directed graph. An *oriented cycle* C of D consists of *forward arcs* C^+ and *backward arcs* C^-, such that $C = C^+ \dot\cup C^-$ and reorienting all arcs in C^- results in a directed cycle. A *directed cycle basis* of D is a set of oriented cycles $C_1, \ldots, C_k$ with incidence vectors $\gamma_i \in \{-1, 0, 1\}^{|A|}$ that permit a unique linear combination of

[1] For $T = 10, \ell_{ij} = 9$, and $u_{ij} = 11$, $\pi_j = 9$ and $\pi_i = 0$ yield $p_{ij} = 0$; $p_{ij} = 2$ is achieved by $\pi_j = 0$ and $\pi_i = 9$.

the incidence vector of any (oriented) cycle of D, where k denotes the cyclomatic number $k = |A| - |V| + 1$ of D. Arithmetic is performed over the field $\mathbb{Q}$.

For a directed graph D, we obtain the *underlying undirected graph* G by removing the directions from the arcs. A *cycle basis of an undirected graph* $G = (V, E)$ is a set of undirected cycles $C_1, \dots, C_k$ with incidence vectors $\phi_i \in \{0,1\}^{|E|}$, that again permit to combine any cycle of G. Here, arithmetic is over the field GF(2). A set of directed cycles $C_1, \dots, C_k$ *projects onto an undirected cycle basis*, if by removing the orientations of the cycles, we obtain a cycle basis for the underlying undirected graph G.

Lemma 1. *Let $\mathcal{C} = \{C_1, \dots, C_k\}$ be a set of oriented cycles in a directed graph D. If $\mathcal{C}$ projects onto an undirected cycle basis, then $\mathcal{C}$ is a directed cycle basis.*

This can easily be verified by considering the $\bmod 2$ projection of $\mathcal{C}$, cf. Liebchen and Peeters[15]. But the converse is not true, as can be seen through an example defined on K_6, with edges oriented arbitrarily ([15]).

Objective Function for Short Cycle Bases. Considering the co-tree arcs in the spanning tree approach as representatives of the elements of a directed cycle basis enables us to formalize the desired property of cycle bases that we need to construct a promising MIP formulation for cyclic timetabling instances.

Definition 1 (Width of a Cycle Basis). *Let $\mathcal{C} = \{C_1, \dots, C_k\}$ be a directed cycle basis of a constraint graph $D = (V, A, \ell, u)$. Let T be a fixed period time. Then, for a_{C_i} and b_{C_i} as defined in (3), we define the* width of $\mathcal{C}$ *by* $W(\mathcal{C}) := \prod_{i=1}^{k} (b_{C_i} - a_{C_i} + 1)$.

The width is our empirical measure for the estimated running time of the MIP solver on instances of the originating application. Hence, for the spanning tree approach, we should construct a spanning tree whose cycle basis minimizes the width function. Especially, if many constraints have small *span* $d_a := u_a - \ell_a$, the width will be much smaller than the general bound $3^{|A|}$, which we deduced from the initial formulation (1) of the PESP. To deal with the product and the rounding operation for computing a_{C_i} and b_{C_i}, we consider a slight relaxation of the width:

$$W(\mathcal{C}) \leq \prod_{i=1}^{k} \left\lceil \frac{1}{T} \sum_{a \in C_i} d_a \right\rceil . \tag{4}$$

De Pina[21] proved that an undirected cycle basis that minimizes the *linearized objective* $\sum_{i=1}^{k} \sum_{a \in C_i} d_a$ also minimizes the right-hand-side in (4). But there are pathological examples in which a minimal cycle basis for the linearized objective does not minimize the initial width function, see Liebchen and Peeters[15].

Applying the above linearization to spanning trees yields the problem of finding a minimal strictly fundamental cycle basis. But two decades ago, Deo et al.[7] showed this problem to be $\mathcal{NP}$-hard. Recently, Amaldi[1] established MAX-SNP-hardness.

General Cycle Bases are Misleading. De Pina[21] keeps an integer variable in the PESP only for the cycles of some undirected cycle bases. Consequently, he could exploit

Horton's[11] $\mathcal{O}(m^3 n)$-algorithm[2] for constructing a minimal cycle basis subject to the linearized objective, in order to find a cycle basis which is likely to have a small width.

In more detail, for a directed cycle basis $\mathcal{C}$, define the *cycle matrix* Γ to be its arc-cycle-incidence matrix. He claimed that the solution spaces stay the same, in particular

$$\{p \in \mathbb{Z}^m \mid p \text{ allows a PESP solution}\} \overset{?}{\subseteq} \{\Gamma q \mid q \in \mathbb{Z}^{\mathcal{C}},\ q \text{ satisfies (2) on } \mathcal{C}\}. \quad (5)$$

We show that, in general, inclusion (5) does *not* hold. Hartvigsen and Zemel[10] provided a cycle basis $\mathcal{C}$ for their graph M_1, cf. Figure 1. For our example, we assume

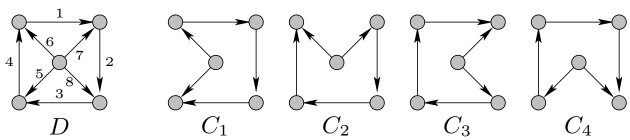

Fig. 1. Cycle basis $\mathcal{C} = \{C_1, \ldots, C_4\}$ for which de Pina's approach fails

that the PESP constraints of D allow only the first unit vector e_1 for p in any solution and choose the spanning tree H with $p|_H = 0$ to be the star tree rooted at the center node.

For $\mathcal{C}$, the transpose of the cycle matrix Γ and the inverse matrix of the submatrix Γ', which is Γ restricted to the rows that correspond to $A \setminus H$, are

$$\Gamma^t = \begin{bmatrix} 1 & 1 & 1 & 0 & -1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & -1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & -1 \end{bmatrix} \quad \text{and} \quad (\Gamma')^{-1} = \frac{1}{3} \begin{bmatrix} 1 & 1 & 1 & -2 \\ -2 & 1 & 1 & 1 \\ 1 & -2 & 1 & 1 \\ 1 & 1 & -2 & 1 \end{bmatrix}.$$

The unique inverse image of $p = e_1$ is $q = (\Gamma')^{-1} p|_{A \setminus H} \notin \mathbb{Z}^k$. Thus, the only feasible solution will not be found when working on $\mathbb{Z}^{\mathcal{C}}$. In the following section we will establish that the crux in this example is the fact that there is a regular $k \times k$ submatrix of the cycle matrix whose determinant has an absolute value different from one.

Thus, key information is lost, when only integer linear combinations of the cycles of some arbitrary cycle basis are considered. To summarize, our dilemma is: Cycle bases over which minimization is easy do not fit our purpose. But minimization over cycle bases that are suitable to formulate instances of cyclic timetabling, becomes $\mathcal{NP}$-hard.

3 Matrix-Classification of Directed Cycle Bases

In order to develop algorithms that construct short cycle bases which we may use for expressing instances of cyclic timetabling, we want to identify an appropriate class of

[2] Golynski and Horton[9] adapted it to $\mathcal{O}(m^s n)$, with s being the exponent of fast matrix multiplication. By a substantially different approach, de Pina[21] achieved a $\mathcal{O}(m^3 + mn^2 \log n)$-algorithm for the same problem.

cycle bases. Fortunately, there is indeed some space left between directed cycle bases that project onto undirected ones, and cycle bases which stem from spanning trees. As our classification of this space in between will be based on properties of cycle matrices, we start by giving two algebraic lemmata.

Lemma 2. *Consider a connected digraph D, with a directed cycle basis $\mathcal{C}$ and the corresponding $m \times k$ cycle matrix Γ. A subset of k rows Γ' of Γ is maximal linearly independent, if and only if they correspond to arcs which form the co-tree arcs of a tree.*

Proof. To prove sufficiency, consider a spanning tree H of D, and $\{a_1, \ldots, a_k\}$ to become co-tree arcs. Consider the cycle matrix Φ with the incidence vector of the unique cycle in $H \cup \{a_i\}$ in column i. As $\mathcal{C}$ is a directed cycle basis, there is a unique matrix $B \in \mathbb{Q}^{k \times k}$ for combining the cycles of Φ, i.e. $\Gamma B = \Phi$. By construction, the restriction of Φ to the co-tree arcs of H is just the identity matrix. Hence, B is the inverse matrix of Γ'.

Conversely, if the arcs that correspond to the $n-1$ rows which are *not* in Γ' contain a cycle C, take its incidence vector γ_C. As $\mathcal{C}$ is a directed cycle basis, we have a unique solution $x_C \neq \mathbf{0}$ to the system $\Gamma x = \gamma_C$. Removing $n-1$ rows that contain C cause x_C to become a non-trivial linear combination of the zero vector, proving Γ' to be singular.

Lemma 3. *Let Γ be the $m \times k$ cycle matrix of some directed cycle basis $\mathcal{C}$. Let A_1 and A_2 be two regular $k \times k$ submatrices of Γ. Then we have* $\det A_1 = \pm \det A_2$.

Proof. By Lemma 2, the k rows of A_1 are the co-tree arcs $a_1, \ldots, a_k$ of some spanning tree H. Again, consider the cycle matrix Φ with the incidence vector of the unique cycle in $H \cup \{a_i\}$ in column i. We know that Φ is totally unimodular (Schrijver[22]), and we have $\Phi A_1 = \Gamma$, cf. Berge[2]. Considering only the rows of A_2, we obtain $\Phi' A_1 = A_2$. As $\det \Phi' = \pm 1$, and as the det-function is distributive, we get $\det A_1 = \pm \det A_2$.

Definition 2 (Determinant of a Directed Cycle Basis). *For a directed cycle basis $\mathcal{C}$ with $m \times k$ cycle matrix Γ and regular $k \times k$ submatrix Γ', the* determinant of $\mathcal{C}$ *is*

$$\det \mathcal{C} := |\det \Gamma'|.$$

We first investigate how this determinant behaves for general directed cycle bases, as well as for those who project onto undirected cycle bases.

Corollary 1. *The determinants of directed cycle bases are positive integers.*

Theorem 2. *A directed cycle basis $\mathcal{C}$ projects onto a cycle basis for the underlying undirected graph, if and only if* $\det \mathcal{C}$ *is odd.*

Due to space limitations, we omit a formal proof and just indicate that taking the $\bmod 2$ projection after every step of the Laplace expansion for the determinant of an integer matrix maintains oddness simultaneously over both, $\mathbb{Q}$ and GF(2).

The following definition introduces the largest class of cycle bases from which we may select elements to give compact formulations for instances of the PESP.

Definition 3 (Integral Cycle Basis). *Let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be cycles of a digraph D, where k is the cyclomatic number $k = |A| - |V| + 1$. If, for every cycle C in D, we can find $\lambda_1, \ldots, \lambda_k \in \mathbb{Z}$ such that $C = \sum_{i=1}^{k} \lambda_i C_i$, then $\mathcal{C}$ is an* integral cycle basis.

Theorem 3 (Liebchen and Peeters[15]). *A directed cycle basis $\mathcal{C}$ is integral, if and only if* $\det \mathcal{C} = 1$.

By definition, for every pair of a strictly fundamental cycle basis and an integral cycle basis with cycle matrices Γ and Φ, respectively, there are unimodular matrices B_1 and B_2 with $\Gamma B_1 = \Phi$ and $\Phi B_2 = \Gamma$. Thus, integral cycle bases immediately inherit the capabilities of strictly fundamental cycle bases for expressing instances of cyclic timetabling. Moreover, the example in Figure 1 illustrates that, among the classes we consider in this paper, integral cycle bases are the most general structure for keeping such integer transformations. Hence, they are the most general class of cycle bases allowing to express instances of the periodic event scheduling problem.

Corollary 2. *Every integral cycle basis projects onto an undirected cycle basis.*

The cycle basis in Figure 1 already provided an example of a directed cycle basis that is not integral, but projects onto an undirected cycle basis.

Theorem 3 provides an efficient criterion for recognizing integral cycle bases. But this does not immediately induce an (efficient) algorithm for constructing a directed cycle basis being minimal among the integral cycle bases. Interpreting integral cycle bases in terms of lattices (Liebchen and Peeters[15]) might allow to apply methods for lattice basis reduction, such as the prominent L^3[13] and Lovász-Scarf algorithms. But notice that our objective function has to be adapted carefully in that case.

4 Special Classes of Integral Cycle Bases

There are two important special subclasses of integral cycle bases. Both give rise to good heuristics for minimizing the linearized width function. We follow the notation of Whitney[24], where he introduced the concept of matroids.

Definition 4 ((Strictly) Fundamental Cycle Basis). *Let $\mathcal{C} = \{C_1, \ldots, C_k\}$ be a directed cycle basis. If for some, resp. any, permutation σ, we have*

$$\forall\, i = 2, \ldots, k : \; C_{\sigma(i)} \setminus (C_{\sigma(1)} \cup \cdots \cup C_{\sigma(i-1)}) \neq \emptyset,$$

then $\mathcal{C}$ is called a fundamental *resp.* strictly fundamental *cycle basis.*

The following lemma gives a more popular notion of strictly fundamental cycle bases.

Lemma 4. *The following properties of a directed cycle basis $\mathcal{C}$ for a connected digraph D are equivalent:*

1. *$\mathcal{C}$ is strictly fundamental.*
2. *The elements of $\mathcal{C}$ are induced by the chords of some spanning tree.*
3. *There are at least k arcs that are part of exactly one cycle of $\mathcal{C}$.*

We leave the simple proof to the reader.

Hartvigsen and Zemel[10] gave a forbidden minor characterization of graphs in which every cycle basis is fundamental. Moreover, if $\mathcal{C}$ is a fundamental cycle basis such that $\sigma = \mathrm{id}$ complies with the definition, then the first k rows of its arc-cycle incidence matrix Γ constitute an upper triangular matrix with diagonal elements in $\{-1, +1\}$. As an immediate consequence of Theorem 3, we get

Corollary 3. *Fundamental cycle bases are integral cycle bases.*

The converse is not true, as can be seen in a node-minimal example on K_8, which is due to Liebchen and Peeters[15]. Champetier[4] provides a graph on 17 nodes having a unique minimal cycle basis which is integral but not fundamental. The graph is not planar, as for planar graphs Leydold and Stadler[14] established the simple fact that every minimal cycle basis is fundamental. To complete our discussion, we mention that a directed version of K_5 is a node-minimal graph having a minimal cycle basis which is fundamental, but only in the generalized sense.

The Venn-diagram in Figure 2 summarizes the relationship between the four major subclasses of directed cycle bases.

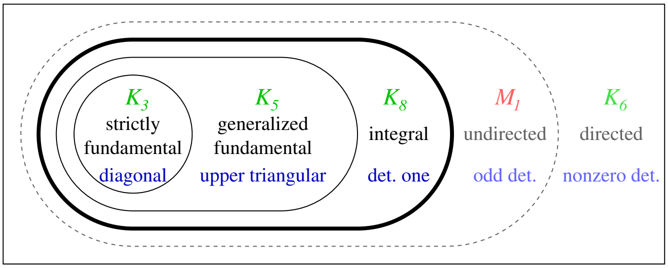

Fig. 2. Map of directed cycle bases

5 Algorithms

A first approach for constructing short integral cycle bases is to run one of the algorithms that construct a minimal undirected cycle basis. By orienting both edges and cycles arbitrarily, the determinant of the resulting directed cycle basis can be tested for having value ± 1. Notice that reversing an arc's or cycle's direction would translate into multiplying a row or column with minus one, which is of no effect for the determinant of a cycle basis. But if our constructed minimal undirected cycle basis is not integral, it is worthless for us and we have to turn to other algorithms.

Deo et al.[6] introduced two sophisticated algorithms for constructing short strictly fundamental cycle bases: *UV* (unexplored vertices) and *NT* (non-tree edges). But the computational results we are going to present in the next section demonstrate that we can do much better. The key are (generalized) fundamental cycle bases. As the complexity status of constructing a minimal cycle basis among the fundamental cycle bases is unknown to the author, we present several heuristics for constructing short fundamental—thus integral—cycle bases. These are formulated for undirected graphs.

Fundamental Improvements to Spanning Trees. The first algorithm has been proposed by Berger[3]. To a certain extent, the ideas of de Pina[21] were simplified in order to maintain fundamentality. The algorithm is as follows:

1. Set $\mathcal{C} := \emptyset$.
2. Compute some spanning tree H with edges $\{e_{k+1}, \dots, e_m\}$.
3. For $i = 1$ to k do
   3.1. For $e_i = \{j, l\}$, find a shortest path P_i between j and l which only uses arcs in $\{e_1, \dots, e_{i-1}, e_{k+1}, \dots, e_m\}$, and set $C_i := e_i \cup P_i$.
   3.2. Update $\mathcal{C} := \mathcal{C} \cup C_i$.

Obviously, the above procedure ensures $e_i \in C_i \setminus \{C_1, \dots, C_{i-1}\}$. Hence, $\mathcal{C}$ is a fundamental cycle basis. Although this procedure is rather elementary, Section 6 will point out the notable benefit it achieves even when starting with a rather good strictly fundamental cycle basis, e.g. the ones resulting from the procedures NT or UV. In another context, similar ideas can be found in Nachtigall[19].

Horton's Approximation Algorithm. Horton[11] proposed *a fast algorithm for a suboptimal cycle basis*. Below, we show that Horton's heuristic always constructs a fundamental cycle basis for a weighted connected graph G.

1. Set $\mathcal{C} := \emptyset$ and $G' := G$.
2. For $i = 1$ to $n - 1$ do
   2.1. Choose a vertex x_i of minimum degree ν in G'.
   2.2. Find all shortest paths lengths in $G' \setminus x_i$ between neighbors $x_{i_1}, \dots, x_{i_\nu}$ of x_i.
   2.3. Define a new artificial network N_i by
      2.3.1. introducing a node s for every edge $\{x_i, x_{i_s}\}$ in G' and
      2.3.2. defining the length of the *branch* $\{s, t\}$ to be the length of a shortest path between x_{i_s} and x_{i_t} in $G' \setminus x_i$.
   2.4. Find a minimal spanning tree H_i for N_i.
   2.5. Let $C_{i_1}, \dots, C_{i_{\nu-1}}$ be the cycles in G' that correspond to branches of H_i.
   2.6. Update $\mathcal{C} := \mathcal{C} \cup \{C_{i_1}, \dots, C_{i_{\nu-1}}\}$ and $G' := G' \setminus x_i$.

Proposition 1. *Horton's approximation algorithm produces a fundamental cycle basis.*

Proof. First, observe that none of the edges $\{x_i, x_{i_s}\}$ can be part of any cycle $C_{r.}$ of a later iteration $r > i$, because at the end of iteration i the vertex x_i is removed from G'. Hence, fundamentality follows by ordering, within each iteration i, the edges and cycles such that $e_{i_j} \in C_{i_j} \setminus (C_{i_1}, \dots, C_{i_{j-1}})$ for all $j = 2, \dots, \nu - 1$. Moreover, every leaf s of H_i encodes an edge $\{x_i, x_{i_s}\}$ that is part of only one cycle. Finally, as H_i is a tree, by recursively removing branches that are incident to a leaf of the remaining tree, we process every branch of the initial tree H_i.

We order the branches $b_1, \dots, b_{\nu-1}$ of H_i according to such an elimination scheme, i.e. for every branch $b_j = \{s_j, t_j\}$, node s_j is a leaf subject to the subtree $H_i \setminus \bigcup_{\ell=1}^{j-1} \{b_\ell\}$. Turning back to the original graph G', for $j = 1, \dots, \nu - 1$, we define e_{i_j} to correspond to the leaf $s_{\nu-j}$, and C_{i_j} to be modeled by the branch $b_{\nu-j}$. This just complies with the definition of a fundamental cycle basis.

6 Computational Results

The first instance has been made available to us by Deutsche Bahn AG. As proposed in Liebchen and Peeters[16], we want to minimize simultaneously both the number of vehicles required to operate the ten given pairs of hourly served ICE/IC railway lines, and the waiting times faced by passengers along the 40 most important connections. Single tracks and optional additional stopping times of up to five minutes at major stations cause an average span of 75.9% of the period time for the 186 arcs that remain after elimination of redundancies within the initial model with 4104 periodic events.

The second instance models the Berlin Underground. For the eight pairs of directed lines, which are operated every 10 minutes, we consider all of the 144 connections for passengers. Additional stopping time is allowed to insert for 22 stopping activities. Hereby, the 188 arcs after eliminating redundancies have an average span of 69.5% of the period time. From earlier experiments we know that an optimal solution inserts 3.5 minutes of additional stopping time without necessitating an additional vehicle. The weighted average passengers' effective waiting time is less than 1.5 minutes.

For the ICE/IC instance, in Table 1 we start by giving the base ten logarithm of the width of the cycle bases that are constructed by the heuristics proposed in Deo et al.[6] These have been applied for the arcs' weights chosen as unit weights, the span $d_a = u_a - \ell_a$, or the negative of the span $T - d_a$. In addition, minimal spanning trees have been computed for two weight functions. The fundamental improvement heuristic has been applied to each of the resulting strictly fundamental cycle bases, For sake of completeness, the width of a minimal cycle basis subject to the linearized objective is given as well. The heuristic proposed by Horton has not been implemented so far.

Subsequently, we report the behavior of CPLEX© [5] when faced with the different problem formulations. We use version 8.0 with standard parameters, except for *strong branching* as variable selection strategy and *aggressive cut generation*. The computations have been performed on an AMD Athlon© XP 1500+ with 512 MB main memory.

Table 1. Influence of cycle bases on running times for timetabling (hourly served ICE/IC lines)

algorithm	global	MST		UV			NT
weight	minima	span	nspan	unit	span	nspan	unit
initial width	34.3	65.9	88.4	59.7	58.6	61.2	58.5
fund. improve	–	41.0	43.2	42.9	42.2	42.9	42.7
without fundamental improvement							
time (s)	–	14720	>28800	20029	23726	6388	>28800
memory (MB)	–	13	113	29	30	10	48
status	–	opt	timelimit	opt	opt	opt	timelimit
solution	620486		667080				629993
fundamental improvement applied							
time (s)	–	807	11985	9305	17963	1103	>28800
memory (MB)	–	1	23	24	30	3	114
status	–	opt	opt	opt	opt	opt	timelimit
solution	–						626051

Due to space limitations, we just summarize that the solution behavior is the same for the instance of the Berlin Underground. The width of a minimal cycle basis is about 10^{39}, and the fundamental improvement reduced the width from values between 10^{62} and 10^{85} down to values ranging from 10^{46} to only 10^{49}. The only computation which exceeded our time limit is again MST nspan without fundamental improvement. Only 19 seconds were necessary to optimize the improved UV nspan formulation.

A key observation is the considerable positive correlation (> 0.44 and > 0.67) between the base ten logarithm of the width of the cycle basis and the running time of the MIP solver. With the exception of only one case, the fundamental improvement either results in a notable speed-up, or enables an instance to be solved to optimality, in case that the time limit of eight hours is reached when not applying the heuristic. Figure 3 provides a detailed insight into the distribution of cycle widths of the basic cycles for the ICE/IC instance before and after the fundamental improvement.

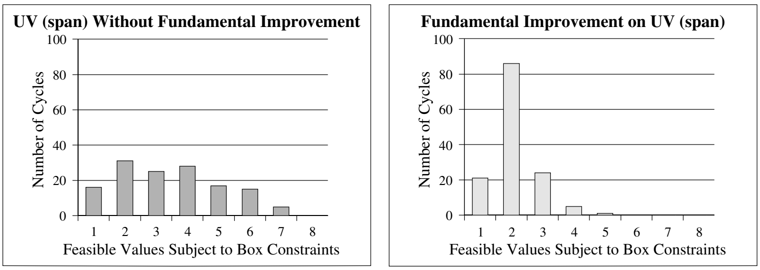

Fig. 3. Shift in distribution of cycle widths due to the fundamental improvements

Since the known valid inequalities, e.g. (2) and Nachtigall[18], heavily depend on the problem formulation, they have not been added in any of the above computations. However, they also provide a major source for improving computation times. For the instance of Deutsche Bahn AG, an optimal solution was obtained after only 66 seconds of CPU time for a formulation refined by 115 additional valid inequalities which were separated in less than 80 seconds.

7 Conclusions

We generalized the standard approach for formulating the cyclic timetabling problem, based on strictly fundamental cycle bases. Integral cycle bases have been established to be the most general class of directed cycle bases that enable the modeling of cyclic timetabling problems. Finally, we presented algorithms that construct short fundamental cycle bases with respect to a reliable empirical measure for estimating the running time of a mixed-integer solver for the originating application.

But some questions remain open. One is the complexity status of minimizing a (linear) objective function over the class of fundamental, or even integral, cycle bases. Another is progress in the area of integer lattices. Finally, it is unknown, whether every graph has a minimal cycle basis that is integral.

Acknowledgments. Franziska Berger, Bob Bixby, Sabine Cornelsen, Berit Johannes, Rolf H. Möhring, Leon Peeters, and of course the anonymous referees contributed in various ways to this paper.

References

1. Amaldi, E. (2003) Personal Communication. Politecnico di Milano, Italy
2. Berge, C. (1962) The Theory of Graphs and its Applications. John Wiley & Sons
3. Berger, F. (2002) Minimale Kreisbasen in Graphen. Lecture on the annual meeting of the DMV in Halle, Germany
4. Champetier, C. (1987) On the Null-Homotopy of Graphs. Discrete Mathematics **64**, 97–98
5. CPLEX 8.0 (2002) `http://www.ilog.com/products/cplex` ILOG SA, France.
6. Deo, N., Kumar, N., Parsons, J. (1995) Minimum-Length Fundamental-Cycle Set Problem: A New Heuristic and an SIMD Implementation. Technical Report CS-TR-95-04, University of Central Florida, Orlando
7. Deo, N., Prabhu, M., Krishnamoorthy, M.S. (1982) Algorithms for Generating Fundamental Cycles in a Graph. ACM Transactions on Mathematical Software **8**, 26–42
8. Gleiss, P. (2001) Short Cycles. Ph.D. Thesis, University of Vienna, Austria
9. Golynski, A., Horton, J.D. (2002) A Polynomial Time Algorithm to Find the Minimum Cycle Basis of a Regular Matroid. In: SWAT 2002, Springer LNCS 2368, edited by M. Penttonen and E. Meineche Schmidt
10. Hartvigsen, D., Zemel, E. (1989) Is Every Cycle Basis Fundamental? Journal of Graph Theory **13**, 117–137
11. Horton, J.D. (1987) A polynomial-time algorithm to find the shortest cycle basis of a graph. SIAM Journal on Computing **16**, 358–366
12. 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, Germany, In German
13. Lenstra, A.K., Lenstra, H.W., Lovász, L. (1982) Factoring polynomials with rational coefficients. Mathematische Annalen **261**, 515–534
14. Leydold, J., Stadler, P.F. (1998) Minimal Cycle Bases of Outerplanar Graphs. The Electronic Journal of Combinatorics **5**, #16
15. Liebchen, C., Peeters, L. (2002) On Cyclic Timetabling and Cycles in Graphs. Technical Report 761/2002, TU Berlin
16. Liebchen, C., Peeters, L. (2002) Some Practical Aspects of Periodic Timetabling. In: Operations Research 2001, Springer, edited by P. Chamoni et al.
17. Nachtigall, K. (1994) A Branch and Cut Approach for Periodic Network Programming. Hildesheimer Informatik-Berichte 29
18. Nachtigall, K. (1996) Cutting planes for a polyhedron associated with a periodic network. DLR Interner Bericht 17
19. Nachtigall, K. (1996) Periodic network optimization with different arc frequencies. Discrete Applied Mathematics **69**, 1–17
20. Odijk, M. (1997) Railway Timetable Generation. Ph.D. Thesis, TU Delft, The Netherlands
21. de Pina, J.C. (1995) Applications of Shortest Path Methods. Ph.D. Thesis, University of Amsterdam, The Netherlands
22. Schrijver, A. (1998) Theory of Linear and Integer Programming. Second Edition. Wiley
23. Serafini, P., Ukovich, W. (1989) A mathematical model for periodic scheduling problems. SIAM Journal on Discrete Mathematics **2**, 550–581
24. Whitney, H. (1935) On the Abstract Properties of Linear Dependence. American Journal of Mathematics **57**, 509–533